LMSC/H/64-001
Cy 3

GPO PRICE $ _____

OTS PRICE(S) $ _____

Hard copy (HC) 2.00

Microfiche (MF) .50

HUNTSVILLE RESEARCH & ENGINEERING CENTER

TECHNICAL REPORT
LMSC/H/64-001
Enclosure (1) to:
LMSC/HREC AO36265
NASA 8-11148

LOCKHEED MISSILES & SPACE COMPANY

HUNTSVILLE RESEARCH & ENGINEERING CENTER

HUNTSVILLE RESEARCH PARK

4800 BRADFORD DRIVE, HUNTSVILLE, ALABAMA

A COMPUTER PROGRAM FOR TRANSIENT THERMAL ANALYSIS OF ARBITRARY SHELLS OF REVOLUTION

May 1964

PREPARED BY: W. D. Whetstone

W. D. Whetstone
Huntsville R & E Center

APPROVED BY: J. S. Farrior

J. S. Farrior
Resident Manager
Huntsville R & E Center

SUMMARY

16316

This report describes a recently developed digital computer program for analyzing transient heat conduction in arbitrary shells of revolution. The program is based on the well-known numerical solution method in which bodies are idealized as assemblages of small "nodes", or volume elements. Familiarity with the foundations of this solution method and some knowledge of FORTRAN are assumed on the part of the reader. Computational techniques employed in the program are discussed in detail and all essential FORTRAN variables are defined. A list of the program and a sample problem execution are included in an appendix (under separate cover). Characterizing features of the program are:

- A very large number of nodes is permissible, allowing representation of bodies as very "fine" nodal networks. The present nominal limit of 5,000 nodes can be expanded significantly without difficulty (most other transient heat conduction programs of which the writer is aware are limited to a few hundred nodes). Considerable emphasis was placed on computational efficiency to make practical (in terms of computer execution cost) the use of very fine nodal networks and their associated relatively short time increments. Slow-access secondary storage (magnetic tape) is not used by the program except for periodic filing of solution data.

- Input data is composed of concise problem definitions, viz: the geometry of the shell, material properties, boundary conditions, and a small number of numerical solution parameters. Non-linear effects (e.g. temperature-dependent material properties, radiation boundary conditions) are allowed. Through a small number of input variables, users of the program control the character of the nodal network, but the task of computing network details (i.e. the number of nodes, individual nodal dimensions, topology, etc.) is performed by the program.

- In the present version of the program, external boundary conditions are specified in the input data in the form of "convective heat rates" (i.e. net heat rates excluding radiant exchange), which are piecewise linear functions of time and location over the exterior surface. However, a program for computing convective rates from trajectory parameters is in a late stage of development at Lockheed's Huntsville Research & Engineering Center. It is planned that, in the near future, the two programs will be combined to form a single highly automatic device. Original general planning of both programs was carried out with this end in view; accordingly, no serious compatibility problems are anticipated.

AUTHO

TABLE OF CONTENTS

1. INTRODUCTION

The computer program described in this report is called the SHORE program (a contraction of Shell of Revolution). It is coded in FORTRAN IV and is presently configured for execution on the IBM 7094. Although oriented primarily toward analysis of aerodynamically heated thin shells of revolution, the general configuration of the SHORE program can accommodate other problem classes, (e.g. plates or rods). Structurally composite shells are allowed, which may be composed of as many as five distinct layers. Layer interface effects (e.g. resistivity discontinuities corresponding to bonding films or air gaps) may be taken into account. Shells are regarded as an assemblage of thin "sections" - cones, cone segments, cylinders, and spherical segments. Each layer of each section may be composed of a distinct homogeneous isotropic material with (generally) temperature-dependent properties.

The present version of the program requires that convective heat input rates* over the exterior surface of the shell be given in the input data as arbitrary piecewise-linear functions of time and location over the body surface. However, a companion program for computing convective heat rates over a wide range of flow conditions is being developed at Lockheed's Huntsville Research & Engineering Center (HREC) and it is planned that the two programs will be coupled to form a single automatic computational device.

Various interior surface boundary conditions are permitted, including "heat sinks". Melting (and ablation representable as melting) is admitted in the outermost layer of the shell.

In addition to two-dimensional (axisymmetric) problems, three-dimensional problems are allowed, provided that one longitudinal symmetry plane exists. Such three-dimensional problems correspond to non-rotating bodies at non-zero angle of attack.

Input data consists of relatively minimum physical problem definitions; viz:

- shell geometry and layer configuration,
- material properties,
- boundary conditions,
- the type of output information desired, and
- certain numerical solution-related parameters.

Using the numerical solution-related parameters appearing in the input, the program establishes the "nodal network" used in approximating solutions to the conduction equation,

$$\nabla k \cdot \nabla T = \rho c \frac{\partial T}{\partial t}, \qquad (1)$$

---

* Since the final draft of this report, the program has been modified to (optionally) accept heat transfer coefficients and recovery temperature as exterior boundary condition input data.

where T = temperature, t = time, k = thermal conductivity, ρ = density, and c = specific heat. Present program capacity is five thousand nodes. This capacity can be considerably expanded without introducing slow-access intermediate storage. In subsequent discussions it will be assumed that the reader is familiar with the usual finite-difference methods* for obtaining approximate solutions to Eq. (1).

* Ref: P. J. Schneider, "Conduction Heat Transfer", Addison-Wesley, Cambridge, Mass., 1955, Ch. 12 (pp. 292-316).

2. THE SHORE PROGRAM

To avoid introducing two complete sets of variables, FORTRAN variable notation will be used almost exclusively throughout the following discussions. Readers who are not familiar with the elements of FORTRAN are referred to the IBM publication: Form C28-6274-1 (FORTRAN IV language).

2.1 Terminology: The terminology presented below will be useful in discussing the class of mathematical models to which the SHORE program applies.

2.1.1 Shell Geometry: Only "thin" shells of revolution are considered: i. e., the total shell thickness must be small compared with minimum radius of curvature at any point. Accordingly, the geometry of the shell is described by specifying the location of the mid-surface generatrix and the layer thicknesses (normal to the mid-surface) as functions of an axial coordinate. The variables X (I), Y (I), NS, and KSP describing mid-surface generatrix location are defined on Figure 1 for a particular example.

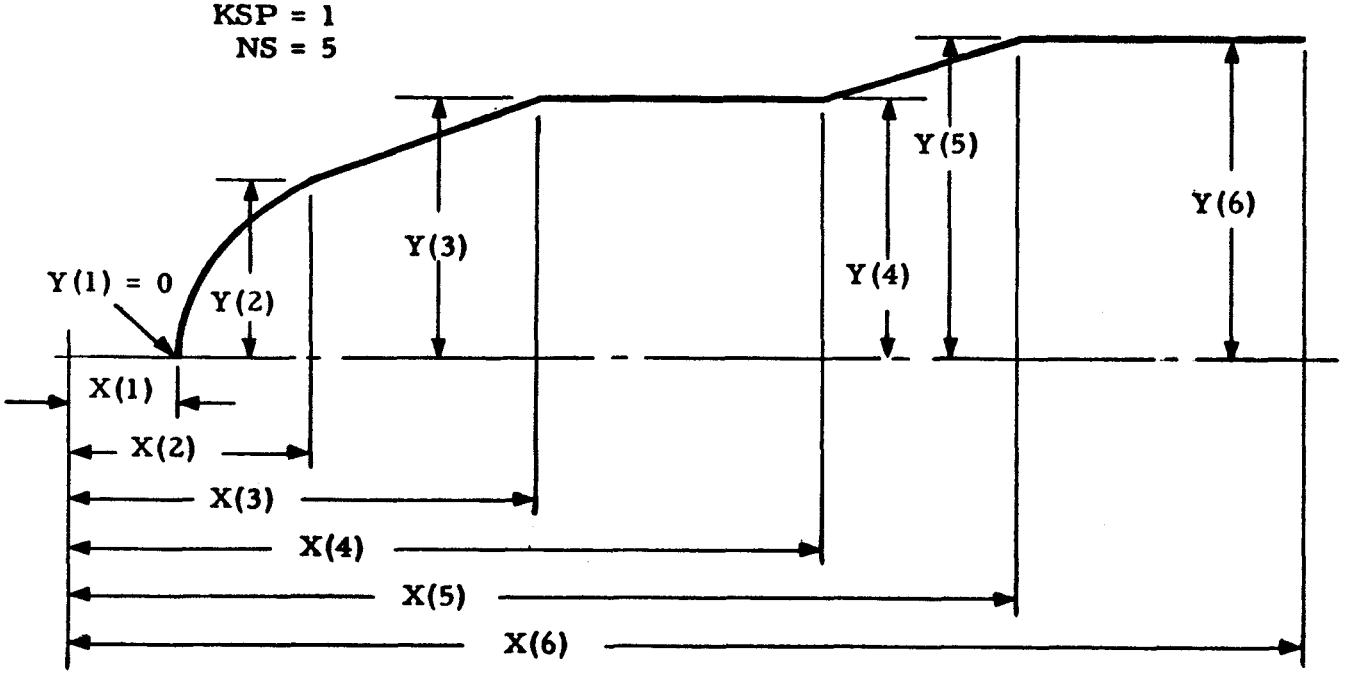


Figure 1

As shown on Figure 1, X (I) and Y (I) are the axial and radial coordinates, respectively, of the "front" of the I-th section. In this case there are five sections. In general, the number of sections is represented by the symbol NS. Accordingly, the range of X (I) and Y (I) is I = 1 through NS + 1. Only the first and/or last sections of a shell may be spherical, as indicated by the value of the variable KSP. KSP is 1, if only the first section is spherical; 2, if only the NS-th section is spherical; and 3, if both the first and NS-th sections are spherical. Other values (e.g. zero) of KSP indicate that neither the first nor the last (NS-th) sections are spherical. The variables NS, and KSP, together with X (I) and Y (I) (for I = 1 through NS + 1) thus completely describe the shell mid-surface. Note that open-ended shells are admitted, i.e., Y (1) and/or Y (NS + 1) may be non-zero.

Linear meridional variations in layer thickness are admitted. As indicated in the example of Figure 2, the variables Z (I, J) represent layer thicknesses throughout the entire shell, except, possibly, within spherical end-sections, which will be discussed later.

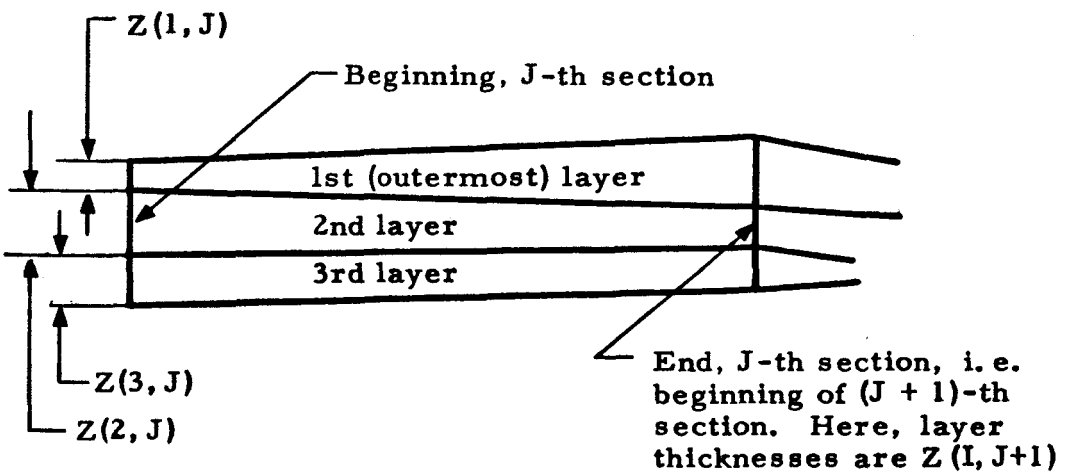
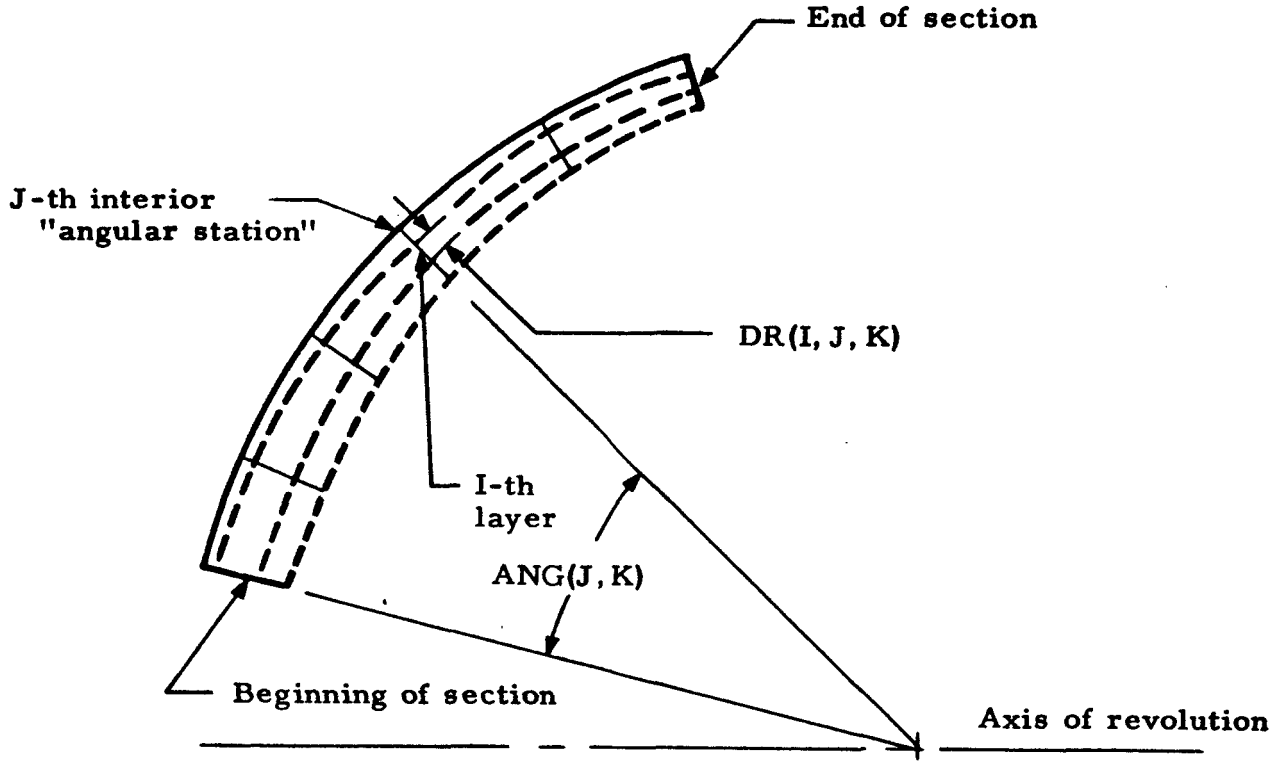


Figure 2

Every section has the same number of layers. The number of layers is represented by the variable NL (on Figure 2, NL = 3). The range of Z (I, J) is I = 1 through NL and J = 1 through NS + 1. As previously indicated, the thicknesses Z (I, J) are dimensions normal to the shell mid-surface. The foregoing definitions ensure that corresponding layers of adjacent sections exactly match, geometrically, at the section boundary. Meridional slope discontinuities (e.g. cone-cylinder junctions) usually present few problems in mathematical modeling of actual shells, provided that the shell is "thin" as previously defined.

Layer thicknesses at the edges of spherical end sections, if any, are included among the previously defined Z (I, J)'s. Also, layer thicknesses at arbitrary "angular stations" interior to such sections may be specified; if linear variations are assumed between all adjacent pairs of such stations. This piecewise-linear meridional variation of layer thickness is specified by the variables DR (I, J, K), ANG (J, K) and NDR (K), as shown in the example of Figure 3.



Cross Section of an "Open" Front Spherical Section

Figure 3

Where K = 1 → nose (first) section and K = 2 → tail (NS-th) section, the location of the J-th "angular station" is given by ANG (J, K) and the corresponding layer thicknesses are DR (I, J, K), for I = 1 through NL. The angles ANG (J, K) are measured (positively) from the beginning of the section toward the end of the section. Where K has the same meaning as defined above, NDR (K) - 2 is the number of interior "angular stations".

All of the variables defined above (NS, X, Y, KSP, NL, Z, DR, ANG, and NDR) are input data to the SHORE program. They completely describe the shell geometry.

2.1.2 Nodal Geometry: The shell is divided into a system of volume elements, or nodes, by three sets of locally orthogonal surfaces. One of these is a system of longitudinal planes passing through the axis of revolution of the shell, specifying tangential nodal subdivisions, as shown on Figure 4. Only half of the shell is considered, since, as previously stated, attention will be confined to problems having a longitudinal symmetry plane.

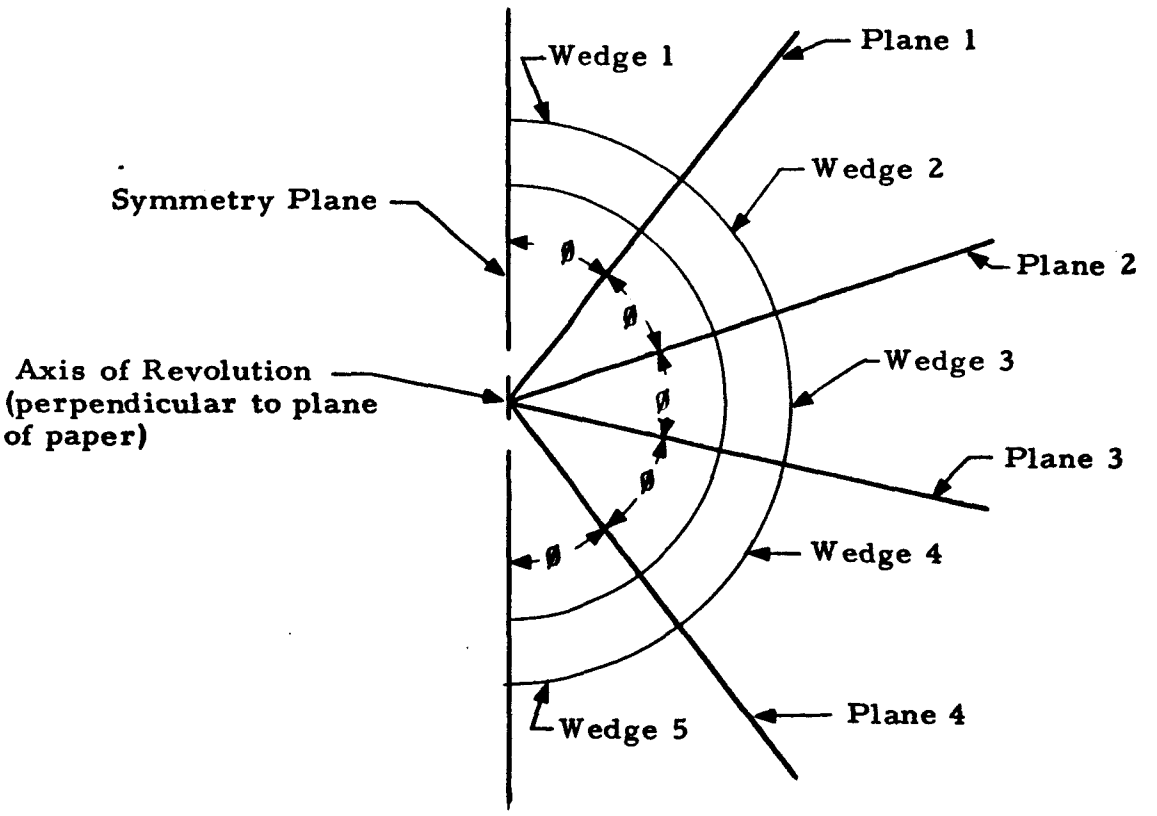


Figure 4

The longitudinal Planes 1 through 4 shown on Figure 4 define all tangential nodal boundaries for a specific example. The collection of nodes contained between an adjacent pair of planes is called a wedge. In the case illustrated, there are five wedges. In general there are NW wedges. The dihedral angles, ϕ, between the longitudinal planes are always equal; i.e. $\phi = \pi/NW$. In the particular case of Figure 4, $\phi = \pi/5$.

The other two sets of nodal boundary surfaces prescribe "radial" and "longitudinal" nodal subdivisions, as shown (for the particular case of a two-layer conical section) on Figure 5.

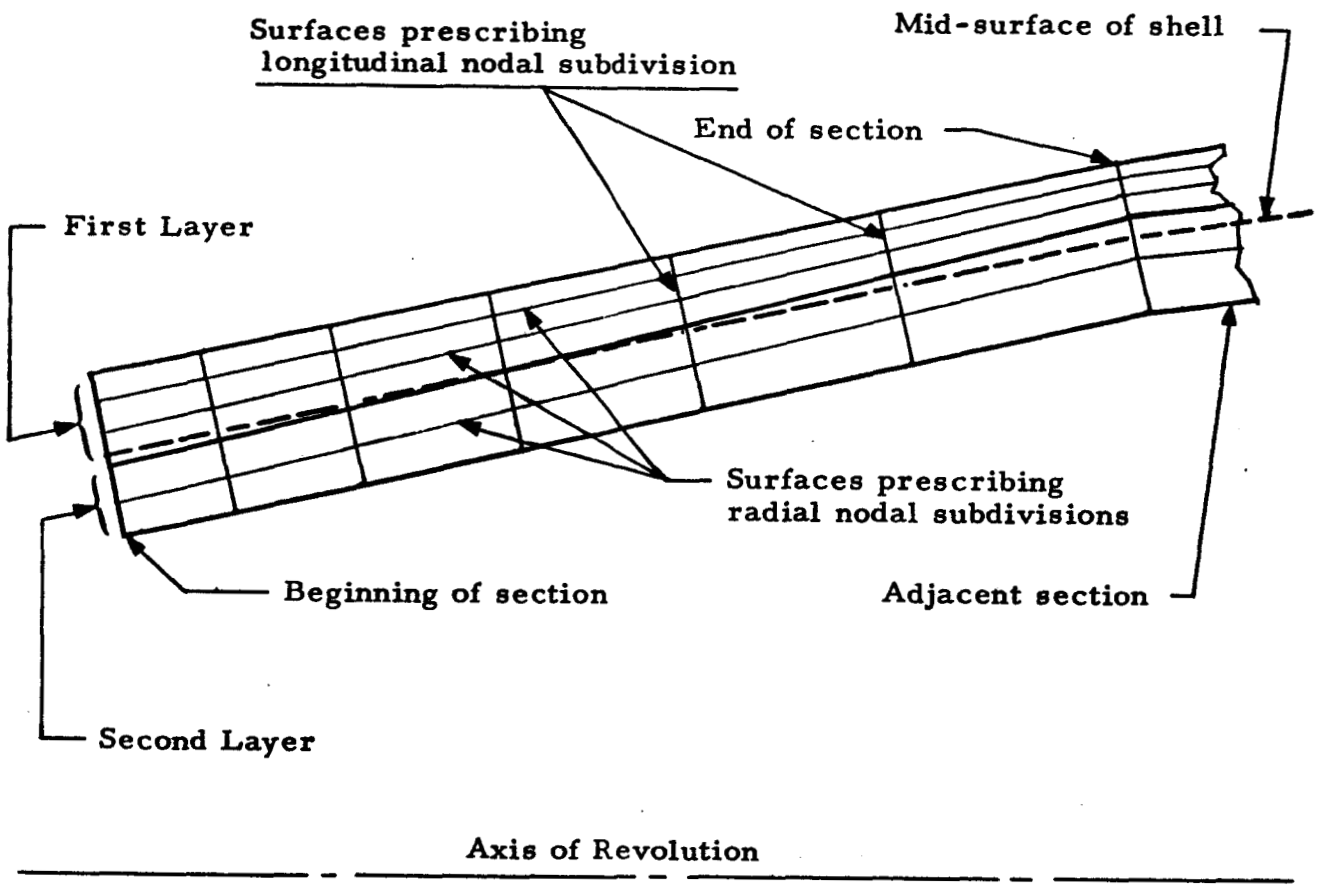


Figure 5

On Figure 5, the plane of the paper coincides with one of the longitudinal planes of Figure 4. Surfaces prescribing longitudinal nodal subdivisions are perpendicular to the shell mid-surface. The collection of nodes contained between an adjacent pair of longitudinal subdivision surfaces, which are generally cones, is called a <u>disk.</u>

The third set of nodal boundary surfaces, the ones prescribing radial nodal subdivisions, are nearly "parallel" to the shell mid-surface. The collection of nodes contained between two such adjacent surfaces is called a <u>band.</u> Throughout all sections, corresponding layers contain the same number of bands. The number of bands in the I-th layer is MB (I). In the example of Figure 5, MB(1) = 3 and MB(2) = 2. The radial * thicknesses of all bands of a given layer are equal; i. e., a line drawn perpendicularly to the shell mid-surface at any point is subdivided into segments of equal length by the radial nodal bounding surfaces within each layer.

Spacing of the longitudinal subdivision surfaces is specified by <u>squareness factors,</u> as defined on Figure 6.

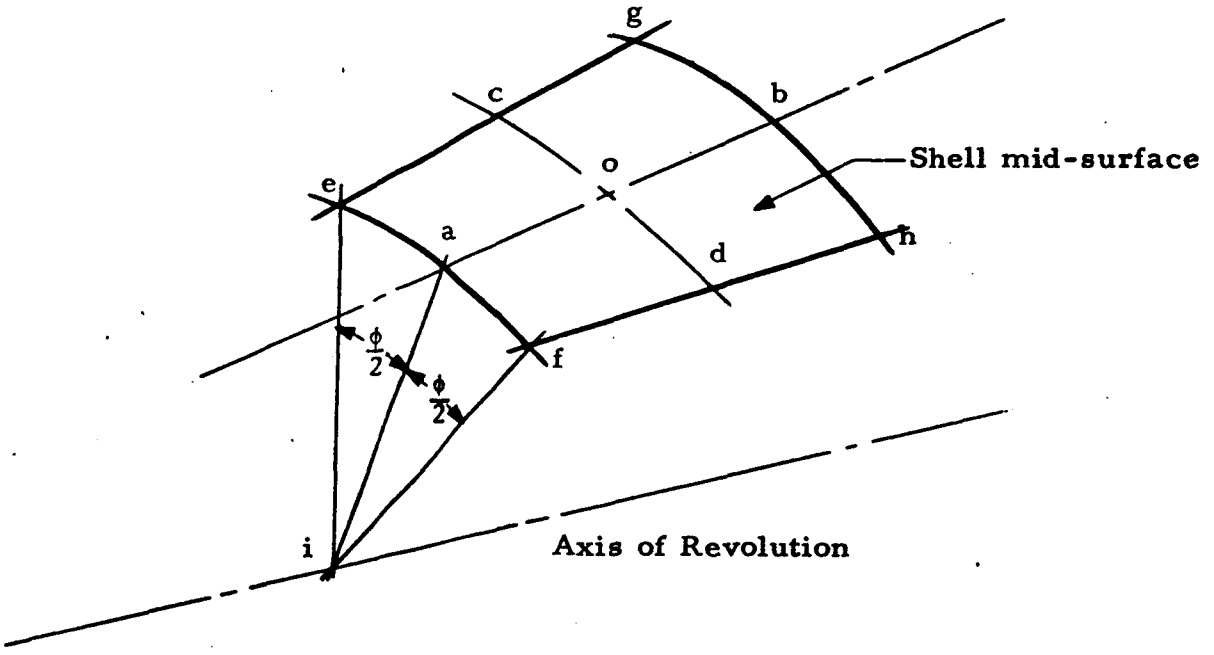


Figure 6

* Here, the term "radial" refers to the direction normal to the shell mid-surface, rather than (necessarily) to directions perpendicular to the axis of revolution.

On Figure 6, curves $\widehat{eg}$ and $\widehat{fh}$ are intersections of two adjacent planes (specifying tangential nodal subdivisions) with the shell mid-surface. These two curves are straight lines within conical or cylindrical sections, and are circular within spherical sections. Circular segments $\widehat{ef}$ and $\widehat{gh}$ are intersections with the mid-surface of two adjacent surfaces (cones) prescribing longitudinal nodal subdivisions. Since attention is confined to "thin" shells, the surface (eghf) is essentially congruent with the radial surfaces of all nodes contained within the two pairs of bounding planes defining points, e, g, h, and f. Points a, b, c, and d are mid points of the boundary curves of (eghf). The center, o, of (eghf) lies midway along the line of intersection, $\widehat{ab}$, of plane (iab) with surface (eghf). The ratio of the arc length of circular segment $\widehat{cod}$ to the length of curve $\widehat{ab}$ is called a nodal squareness factor. Disregarding the "end nodes" at the extremities of terminal spherical or conical sections, it is possible that the squareness factors of all nodes in each section may be made equal by properly choosing the locations of the longitudinal nodal subdivision surfaces (assuming that NW, hence the tangential boundaries of all nodes, is given). For example, consider the cylindrical section shown on Figure 7.

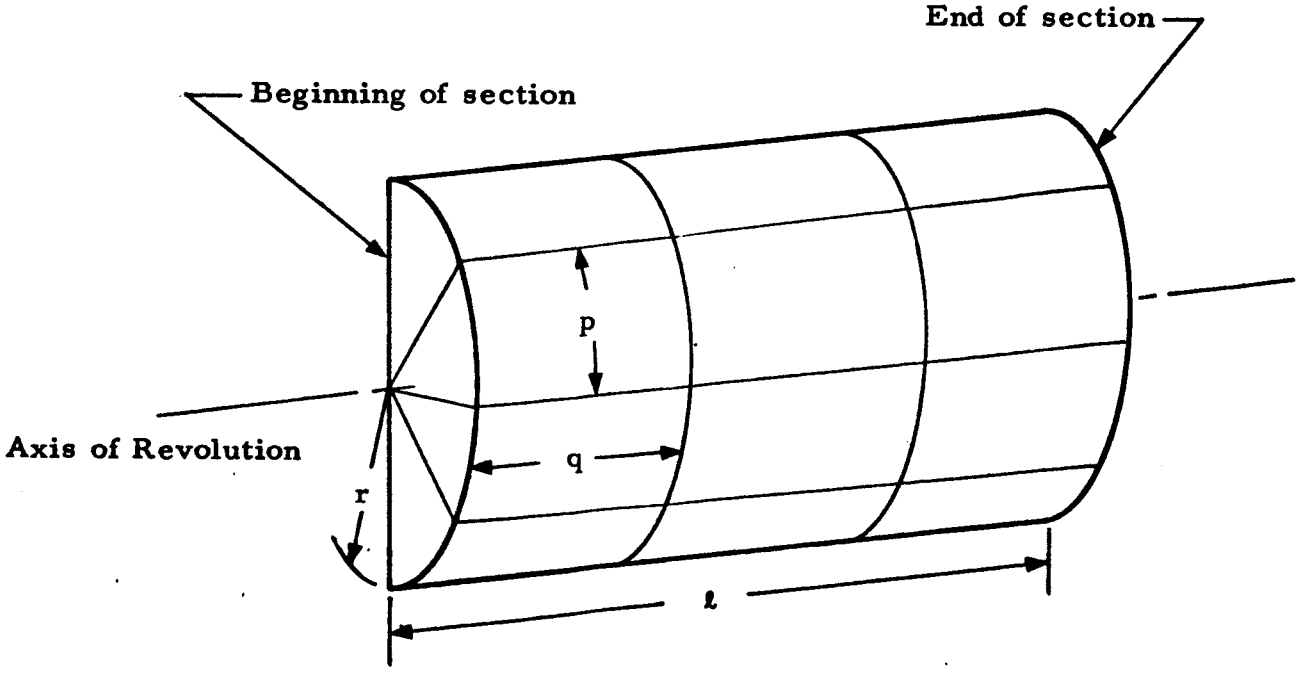


Figure 7

In this example NW is given as 4 so that the circumferential "width", p, of each node is $\pi r/4$. Corresponding to the three equally spaced longitudinal nodal subdivisions, the "length", q, of each node is $\ell/3$; and the squareness factor is $(3\pi r)/4\ell$. If this cylinder were divided into n equally spaced bands (rather than 3), the squareness factor of each node would be $(n\pi r)/(4\ell)$. Criteria for the equality of squareness factors within sections composed of conical segments are discussed below.

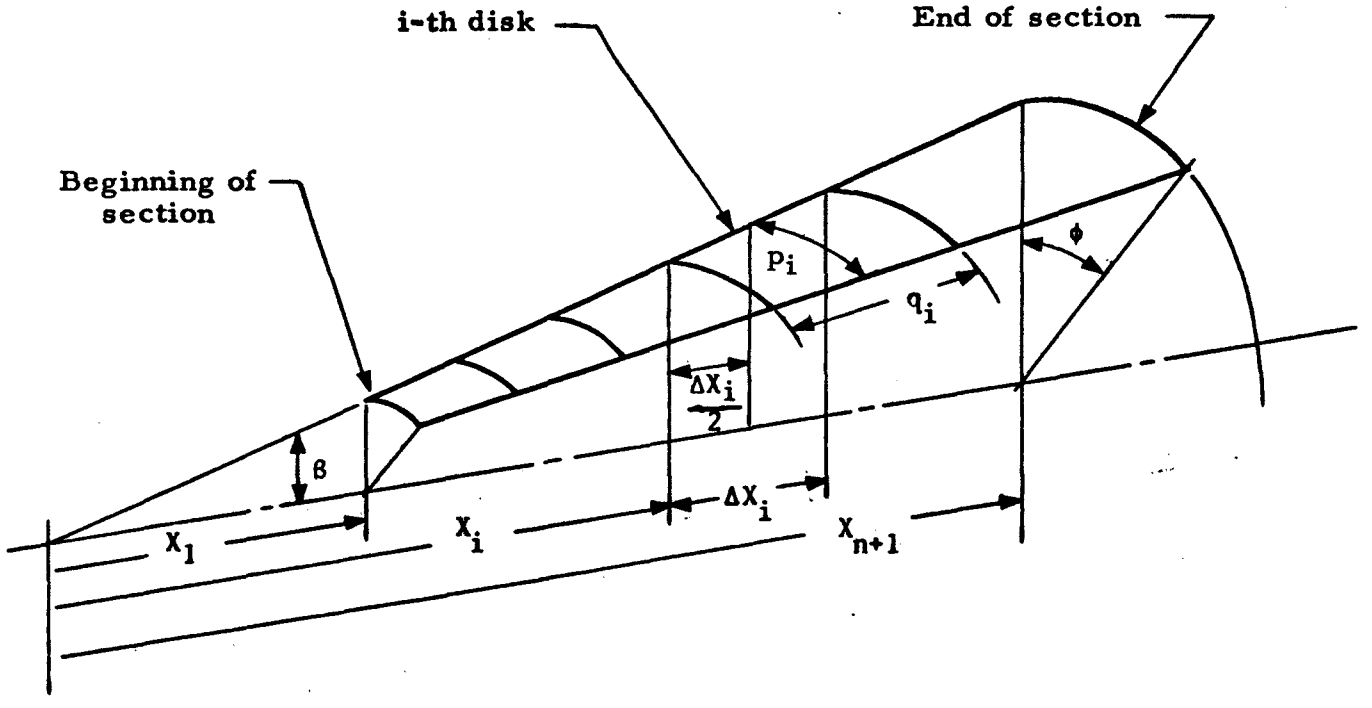


Figure 8

The cone-segment section shown on Figure 8 is divided into n disks. Since

$$p_i = \phi(X_i + 1/2\ \Delta X_i)\ \text{Tan}\ \beta, \text{ and}$$

$$q_i = \frac{\Delta X_i}{\text{Cos}\ \beta};$$

the squareness factor, f, for i = 1 through n is

$$f \equiv \frac{p_i}{q_i} = \phi \sin \beta\ \frac{X_i + 1/2\ \Delta X_i}{\Delta X_i}$$

that all squareness factors are very nearly equal to the nominal value; however, in this case, the actual squareness factors vary slightly from disk to disk. Disk boundary locations within cones and cylinders are computed such that all nodes of each section have identical squareness factors; the squareness factor for nodes of each section being equal to, or barely greater than the stated nominal value.

As part of the data output from the SHORE program the actual (computed) squareness factors for each conical and cylindrical section, if any, are given; as are those of each disk of spherical end-sections, if present. The number of disks in the I-th section, as computed by the program, is MD (I).

The geometry of the nodal network is completely determined by the program from the input variables MB, NW, and SQ.

2.2 Numerical Solution Procedure: In numerically approximating solutions to Eq. (1), shells are idealized as an assemblage of nodes as defined in Section 2.1. Based on the boundary conditions and node temperatures existing at a given time, it is assumed that heat transfer rates through all node faces are essentially constant during a succession of "small" time increments. Assuming that the time increments (hence the changes in node temperatures) are sufficiently small that the properties (e.g. specific heat) of all nodes are essentially unchanged during each time interval, it is possible to compute all changes in nodal temperatures during the interval.

The process employed by the SHORE program in carrying out this solution procedure is discussed below.

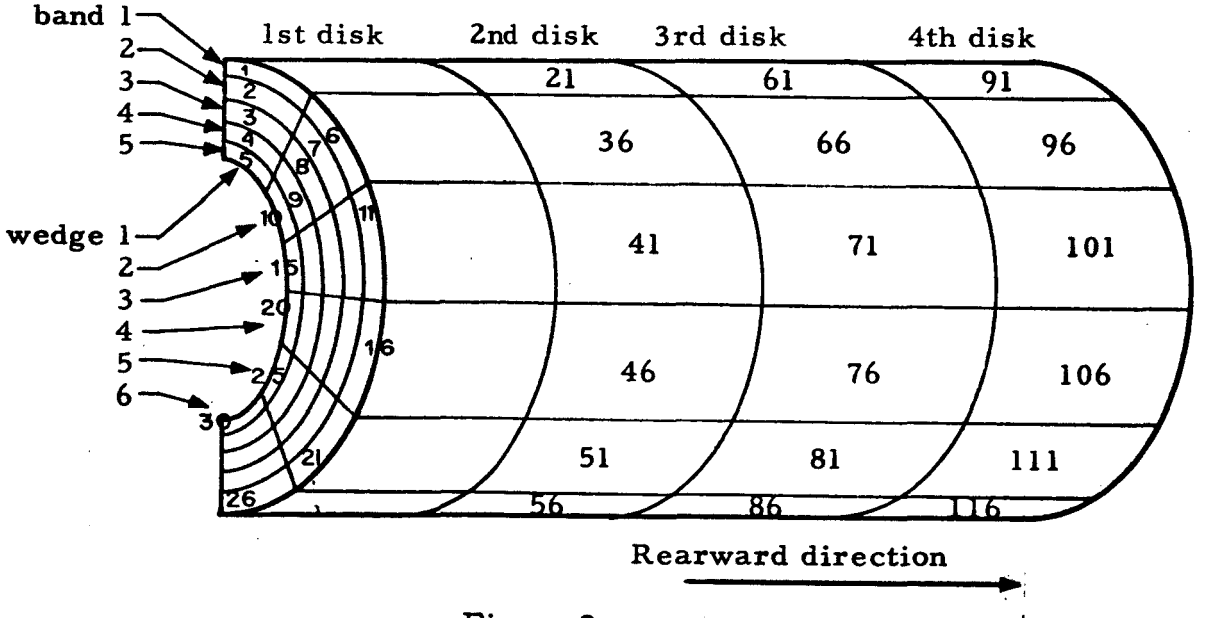


Figure 9

For the present, attention will be confined to open-ended shells such as the one shown on Figure 9. "Closed" shells will be discussed later. Nodes are numbered (indexed) according to their position within the nodal network, as follows:

- Within a given wedge and disk, nodes are numbered in ascending serial order; proceeding from the outermost band toward the innermost.
- Serial order indexing proceeds from one wedge to the next (positively in the clockwise direction looking rearward along the axis of revolution) within the same disk.
- Indexing continues in serial order from the last node of each disk to the first node of the following disk.

Note that the number of nodes in each wedge of all disks is the same; viz. NN, where

$$NN \equiv \sum_{I=1}^{NL} MB(I)$$

The FORTRAN variable NPD represents the total number of nodes per disk; i. e. NPD = NW times NN.

In the example of Figure 9, NN = 5 (the shell might consist of two layers, one containing two bands; the other, three), NW = 6, and NPD = 30.

At the beginning of each time interval, all nodal temperatures are stored in an array, T; i. e., the temperature of the I-th node is T(I). In computing the "new" temperatures at the end of a time increment, the SHORE program proceeds as follows:

First, the heat transfer through all faces of every node in the first disk is computed. The net heat transfer into each node is stored in an array, U; i. e. U(I) is the net heat transferred to the I-th node. All "axial" heat transfer terms from each node of the first disk into the adjacent node in the second disk are stored in an array, QA. Then, using the array U, "new" temperatures are computed for each node in the first disk. These new temperatures are stored in exactly the same locations as were the corresponding temperatures at the beginning of the time increment, since the immediately available array of axial heat transfer terms, QA, precludes the necessity for retaining the original temperatures in the first disk. The same process is executed in the second disk, then the third, and so on to the end of the shell. The arrays U and QA are used over and over again.

Data storage resources are used very efficiently, compared with the storage arrangement which would be required by the alternative of maintaining two distinct temperature arrays: one for temperature at the beginning of a time interval and another for the "new" temperatures.

2.2.1 Heat Conduction Terms: Three classes of heat conduction terms can be conveniently identified: radial, tangential, and axial; which correspond to heat transfer across radially, tangentially, and axially adjacent nodal surfaces, respectively. In each case, the conduction rate is obtained by multiplying the corresponding temperature difference into a "lumped" conductivity term. Considering the three nodes bounding the I-th node on its inner radial, clockwise tangential, and rearward axial faces, respectively, the temperature difference terms are:

- Radial: T(I) - T(I+1).
- Tangential: T(I) - T(I+NN).
- Axial: T(I) - T(I+NPD).

These expressions apply, of course, only if there exist corresponding adjacent nodes. Except for the special cases of heat transfer through nodal surfaces coinciding with section or layer interfaces, the "lumped" conductivities for each of the three types of terms have the same form; viz: the thermal conductivity of the material multiplied by the area of the nodal surface under consideration, divided by the distance between the node "centers". In computing such "lumped" conductivities, the SHORE program uses thermal conductivity based on the temperature T(I). The geometrical factors (areas divided by distances) are represented in the SHORE program by the FORTRAN symbols RA (radial) TA (tangential), and AX (axial), which are constructed from the squareness factors, radial nodal thicknesses, etc. Approximations other than those employed by the SHORE program in evaluating these geometrical factors might have been chosen (e. g. use of node centroids as nodal "centers"; use of geodesics in evaluating distances between node centers, etc.). However, it is problematical whether more complex approximations might improve solution accuracy (for a particular nodal network). Similarly, "average" values of thermal conductivity along the paths between node centers might have been used. However, slight refinements such as these might result in substantial penalties in computer-execution time. Related questions regarding the entire numerical solution process are discussed in Section 3, Comments.

In computing axial and radial conduction rates through nodal surfaces coinciding with section or layer interfaces, respectively, the "lumped" conductivities are computed by taking the reciprocal of the sum of the "lumped" thermal resistances of the half-paths of the two nodes. In this case, since the two nodes may possibly consist of different

materials, thermal conductivities for the two half-paths are evaluated at the corresponding node centers. An additional resistance term corresponding to finite inter-layer resistivity "films" appears, if applicable, in the "lumped" thermal resistance for radial conduction.

2.2.2 Closed Terminal Sections: If a terminal section (i. e. the first or last section) of a shell is "closed", a terminal "cap" of nodes appears, as illustrated for a spherical "nose" section on Figure 10.

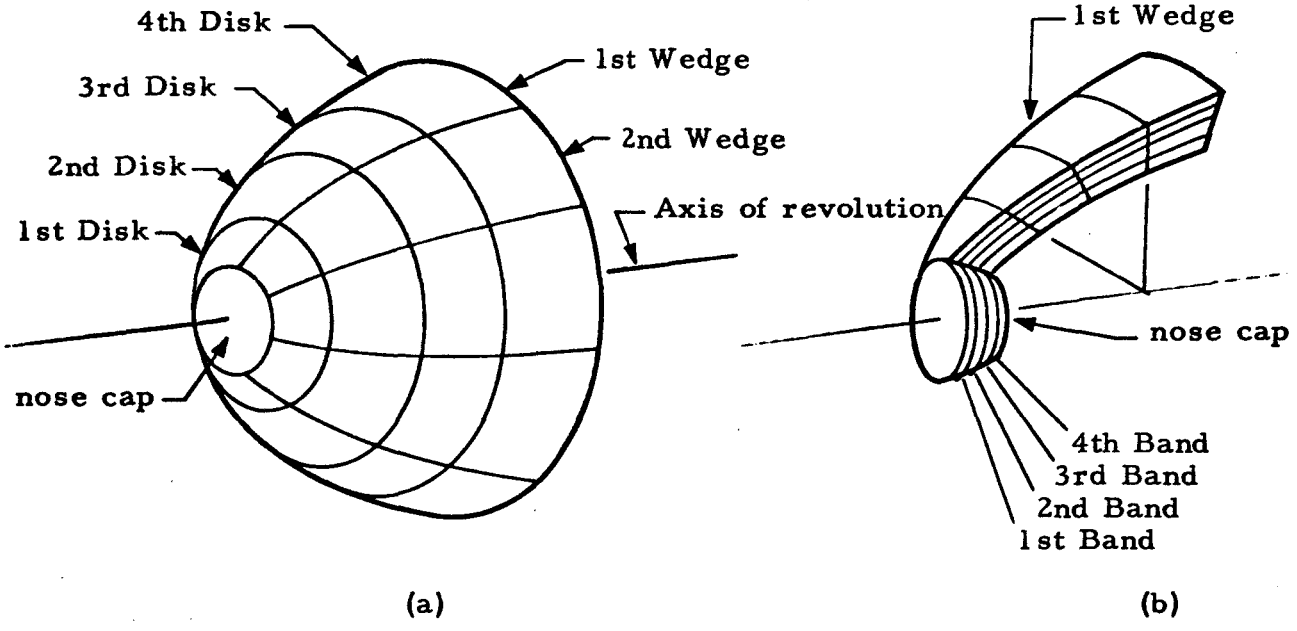


Figure 10

Figure 10 (a) shows the nodal boundary lines as they appear on the exterior shell surface. Figure 10 (b) is a cut-away view showing the arrangement of interior nodal boundaries. Forward axial faces of all nodes in the i-th band in the first disk are bounded by the axial face of the i-th node in the cap. If a nose cap is present, the temperatures of its nodes are T(1), T(2) --- T(NN). Nose cap nodes are assigned the lowest index numbers. "Tail cap" nodes of closed-terminal sections are assigned the highest index numbers. In each case, serial indexing proceeds from the outermost node (1-st band) toward the interior.

In setting up the nodal network, the SHORE program computes terminal cap boundaries such that the radius of the cap nodes is less than one-tenth of the circumferential dimension of the nodes at the opposite end of the section. For purposes of computing conduction terms, etc., the cap nodes are treated as circular plates, having essentially the same radial surface area as the cap nodes themselves.

2.2.3 Surface Layer Melting: In describing the method used by the SHORE program to compute degradation of the outer layer by melting, it is convenient to define surface layer elements. Each disk has NW surface layer elements (one for each wedge), each of which consists of that part of the outer layer bounded by adjacent longitudinal and tangential nodal subdivision surfaces.

For example, suppose that the outer layer of the shell shown on Figure 9 has two bands (i. e. NB(1) = 2). Then the first surface layer element is composed of Nodes 1 and 2, the second of Nodes 6 and 7, and so forth (there is a total of 24 surface elements in this particular example; the last surface layer element consists of Nodes 116 and 117). Surface layer element thicknesses are individually stored in an array, S; the thickness of the I-th element being S(I). If, during a particular time increment, material is melted from the I-th surface element, the value of S(I) is altered accordingly. The radial thicknesses of all nodes within the I-th surface layer element are always equal, although the thickness of the entire layer may change. This amounts to a radial distortion of the nodal network, as shown on Figure 11.

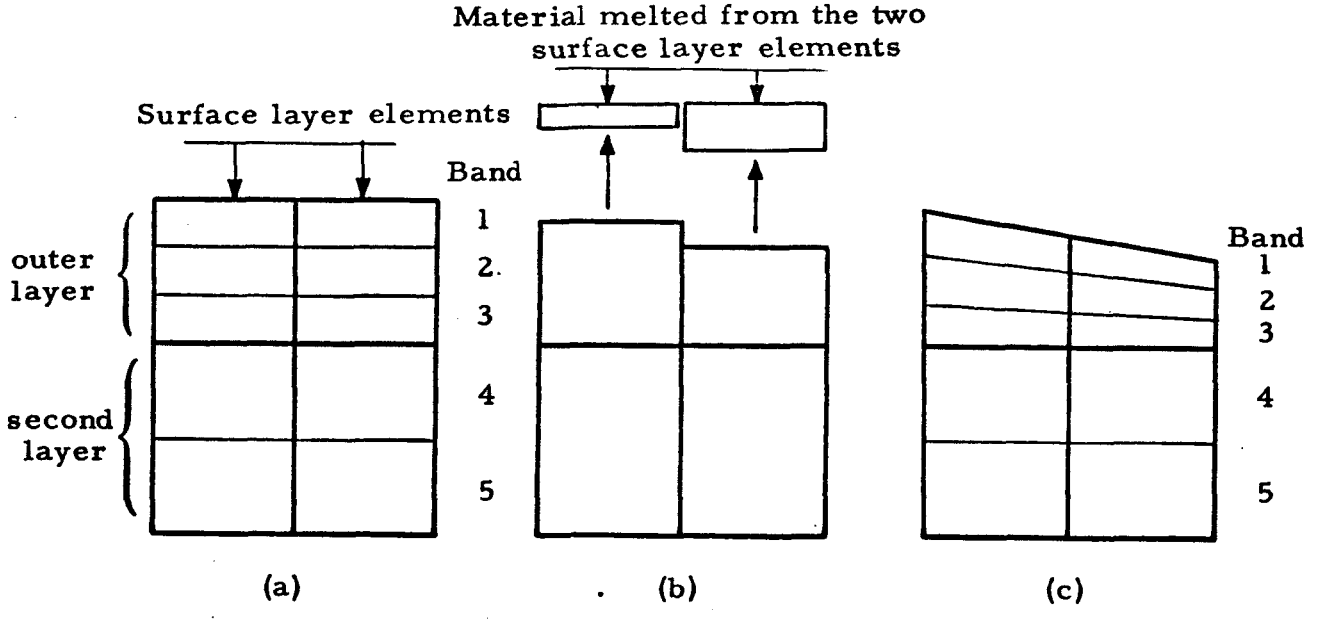


Figure 11

In the example shown on Figure 11 there are two layers. The outer layer has three bands, the second layer has two. Interior radial nodal boundaries at the beginning and end of a time increment are shown on Figures 11 (a) and 11 (c), respectively; corresponding to surface layer element material losses resulting from melting during the time interval, as shown on Figure 10 (b). In order for this process to yield satisfactorily accurate results, time increment length should be sufficiently small that the volume of material melted from a surface layer element during the time increment is small compared with the volume of a node within the element. If the amount of melting varies rapidly (i. e. as a function of location) over a given region of the surface, the axial and circumferential dimensions of the nodes within that region must be comparatively small;* however, it is expected that such will not be the case in most applications.

2.2.4 Axisymmetric Problems: If there are no circumferential temperature gradients (e. g. when the body is aerodynamically heated while flying at zero angle-of-attack), only a single wedge of nodes needs to be considered. Part of the data input to the SHORE program is a "control" variable, KAX, which indicates whether or not the problem is axisymmetric. The input variable NW (the number of wedges) must be stated for all problems, whether or not they are axisymmetric, to enable the program to construct a nodal network having the desired dimensions, spacing, etc. In effect, the SHORE program merely disregards tangential heat conduction terms and confines attention to a single wedge of nodes in analyzing axisymmetric problems. The appearance of the nodal network in a typical case is shown on Figure 12.

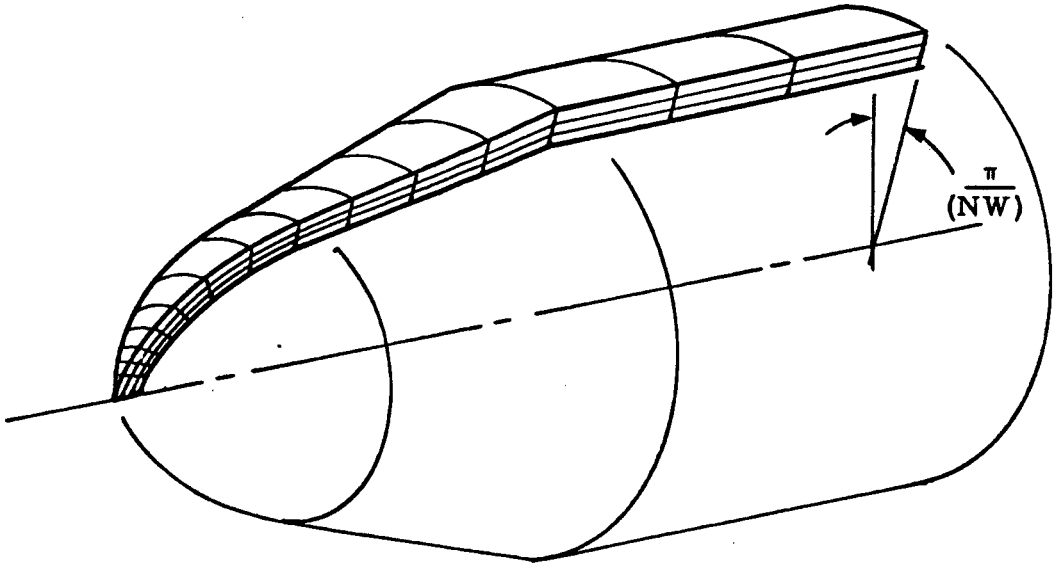


Figure 12

* This question is discussed further in Section 3, Comments.

2.3 Definitions of FORTRAN Variables: As indicated by the flow charts presented in Section 2.4, all of the data input to the SHORE program is read by the subroutine named PREA. In Section 2.3.1, all of the input variables are defined. Implicit in these definitions are the specific capabilities and limitations of the SHORE program regarding boundary conditions, material properties, etc.; so that Section 2.3.1, together with a list of the input routine PREA, provide all information required to prepare data decks for executing specific problems.

FORTRAN variables "originating" elsewhere in the program are defined in Section 2.3.2. Some of the variables defined in previous discussions are re-stated in the following section to provide a complete list.

2.3.1 Input Variables:

HOL (_): Alphameric problem heading statement (not actually part of the problem definition).

DT: Time increment for the first solution of the problem. The time increment does not vary with time. (seconds)

TINIT: Initial temperature, constant throughout body. (^{o}R)

TO4: Ambient temperature for purposes of computing radiant heat input to the body surface. (^{o}R)

NDT: The number of solutions which are to be executed for the stated problem. If NDT = 1, only one solution is executed. If NDT = 2, the time increment is halved and a second solution executed; etc.

The variables KPR, MKPR, IND, NINST (_), NCHRON(_), KORIN (_, _, _), and KORCH (_, _, _) control the production of output. There are two kinds of output data, instantaneous profiles and chronological records.

The instantaneous profiles are printed at regular intervals during the solution process. They consist of lists of adjacent nodal temperatures or surface element thicknesses at specific times. NINST (1) and NINST (2) are the number of temperature and surface element thickness profiles, respectively, to be included in the output. There are three types of instantaneous profiles: radial, tangential, and longitudinal. The "instantaneous profile coordinates", KORIN (_, _, _), specify the details of the desired profiles. Where K = 1 for temperature profiles and K = 2 for surface layer element thickness profiles, the details of the J-th profile are specified by five integers, KORIN (I, J, K) (for I = 1 through 5); as follows:

KORIN (5, J, K): Specifies the type of profile. Possible values are the integers 1, 2, and 3; corresponding to radial,

tangential and longitudinal profiles, respectively.

KORIN (1, J, K): Identifies the section in which the profile occurs; i. e. it is in the KORIN (1, J, K)-th section.

KORIN (2, J, K): Indicates the band in which the profile occurs (i. e. any desired band between 1 and NN).

KORIN (3, J, K): Indicates the wedge in which the profile occurs (i. e. any desired wedge between 1 and NW).

KORIN (4, J, K): Enables the program to select the disk in which the profile occurs. It indicates in per cent (i. e. an integer between 0 and 100) the ratio of the meridional distance between the front of the section and the center of the desired disk location to the meridional "length" of the entire section. The program then selects the disk whose center falls closest to the specified location. For example, if, in the case of the cylinder shown on Figure 9, KORIN (4, _, _) is given as 40, the second disk would be selected by the program. This somewhat irregular method of specifying disk locations was chosen because users will not necessarily know the exact locations of disks before the solution is executed.

Suppose, for example, that a shell is composed of a single cylindrical section as shown on Figure 9, and that six instantaneous profiles (four temperature profiles and two surface element thickness profiles) are desired. In this case, NINST (1) and NINST (2) would be set equal to 4 and 2, respectively. Typically, the details of these profiles might be indicated by the following values of instantaneous profile coordinates:

Temperatures:

Profile Number (J)	KORIN (1, J, 1)	KORIN (2, J, 1)	KORIN (3, J, 1)	KORIN (4, J, 1)	KORIN (5, J, 1)
1	1	-	1	10	1
2	1	-	2	90	1
3	1	4	-	10	2
4	1	1	5	-	3

Surface Element Thicknesses:

Profile Number (J)	KORIN (1, J, 2)	KORIN (2, J, 2)	KORIN (3, J, 2)	KORIN (4, J, 2)	KORIN (5, J, 2)
1	1	-	2	-	3
2	1	-	-	70	2

The specification listed above would result in the output data described on the following page.

Temperatures:

Profile No.	Nodes Indicated
1	Nodes 1 through 5 - a radial profile in wedge 1 of disk 1.
2	Nodes 96 through 100 - a radial profile in wedge 2 of disk 4 (closest to 90% station).
3	Nodes 4, 9, 14, 19, 24, and 29 - a tangential profile through band 4 of disk 1 (closest to 10% station).
4	Nodes 21, 51, 81, and 111 - a longitudinal profile through band 1 of wedge 5.

Surface Layer Element Thicknesses:

Profile No.	Outermost node of surface element indicated
1	Nodes 6, 36, 66, and 96 - a longitudinal profile along wedge 2.
2	Nodes 61, 66, 71, 76, 81, and 86 - a tangential profile around disk 3 (closest to 70% station).

Note that for each type of profile, certain individual "instantaneous profile coordinates" are unnecessary or do not apply. There can be no radial profile of surface element thicknesses. Disk locations need not (in fact cannot) be specified for longitudinal profiles, etc. In axisymmetric problems, it is unnecessary to specify wedge locations for any kind of profile, since the program understands that there is but a single wedge.

The chronological records of temperatures and surface element thicknesses at specific locations are printed after solution is complete. The number of locations for which such records are required is specified by NCHRON (K); where K = 1 → nodal temperatures and K = 2 → surface element thicknesses. For I = 1 through 4, KORCH (I, J, K) indicates the location of the J-th point (node or surface element, according to whether K is 1 or 2, respectively); as follows:

KORCH (1, J, K): section number

KORCH (2, J, K): band number

KORCH (3, J, K): wedge number

KORCH (4, J, K): disk location indication of the same type as KORIN (4, _, _).

The time spacing between entries in the chronological records is indicated by KPR; i. e. entries are made every KPR time increments. The variable MKPR indicates the relative spacing between chronological record entries and instantaneous profile output. For example, if DT = 10.0, KPR = 2, and MKPR = 5, the time spacing between chronological record entries is 20.0 seconds; and instantaneous profiles are printed every 100.0 seconds. The variable IND indicates the time units to be used in identifying the output data. IND is set equal to 1, 2, or 3 according to the units desired: seconds, minutes, or hours, respectively: Note that DT is always given in the input data in seconds (Ft - lb - sec - ^{o}R - BTU input units are standard), and that IND merely controls the output units.

Other "problem definition" variables are:

NS: Number of sections.

NL: Number of layers.

NW: Number of wedges (tangential nodal subdivisions for the half-shell).

KSP: Control variable indicating whether or not the first and/or last sections are spherical; as defined in detail in Section 2.1.1.

KAX: Control variable indicating (if identically zero) axisymmetric problems. Non-zero values indicate non-axisymmetric problems.

MB(I): Number of bands in the I-th layer.

MAT(I, J): Identifying index assigned to the material of which the I-th layer of the J-th section is composed.

NMAT: The number of different materials contained in the shell.

NRES: The number of non-zero inter-layer resistivity "films".

Material property data is specified through the variables HOL, DEN, AB, HF, and P. Each different material is assigned a unique identifying index number. For the K-th material,

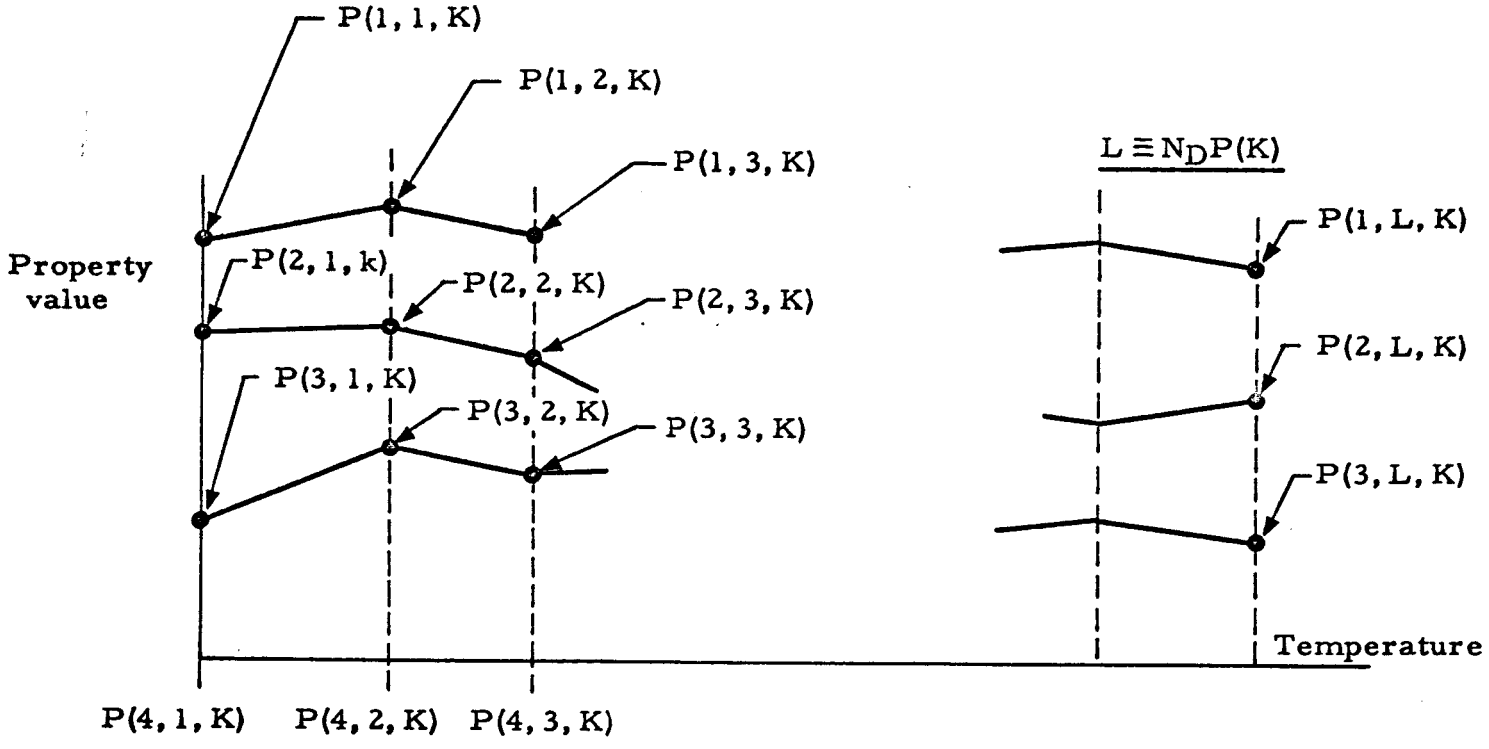
DEN(K): density (lbs/ft^3).

AB(K): melting temperature (^{o}R).

HF(K): heat of fusion (BTU/lb).

HOL(_): alphameric information (e. g. the word ALUMINUM) - eighteen characters are allowed.

Specific heat, thermal conductivity, and emissivity are specified as piecewise-linear functions of temperature as shown on Figure 13.



(K-th Material)

Figure 13

P(1, J, K): Thermal conductivity (BTU/Ft-Sec-^{o}R).

P(2, J, K): Specific heat (BTU/Lb-^{o}F).

P(3, J, K): Emissivity (dimensionless).

P(4, J, K): Temperature (^{o}R) at which the property values P(I, J, K), for I = 1, 2, and 3, apply.

RES(I, J): Resistivity of film between the I-th and (1+I)-th layers of the J-th section (^{o}R-Ft^{2}-Sec/BTU).

SQ(I): Squareness factor of I-th section, as defined in 2.1.2 (dimensionless).

X(I): Axial coordinate, front of I-th section (feet).

Y(I): Radius of revolution, front of I-th section (feet).

Z(I, J): Thickness of I-th layer at the front of the J-th section (feet).

NDR(K), ANG(J, K), DR(I, J, K): Spherical layer thickness data as defined in detail in 2.1.1. (ANG - radian, DR - feet).

INBC: Internal boundary condition control variable. If set equal to zero, it indicates that the interior boundary is insulated. If set equal to 1, it indicates that a single heat sink is adjacent to the entire interior surface. If equal to neither 0 nor I, it indicates that a distinct heat sink is adjacent to the inner boundary of each individual section.

For the I-th heat sink,

TIN (I): Initial temperature (^{o}R).

CSEC (I): Resistivity at edge of sink (^{o}R-Sec-Ft^2/B).

HSEC (I): Heat capacity (B/^{o}R).

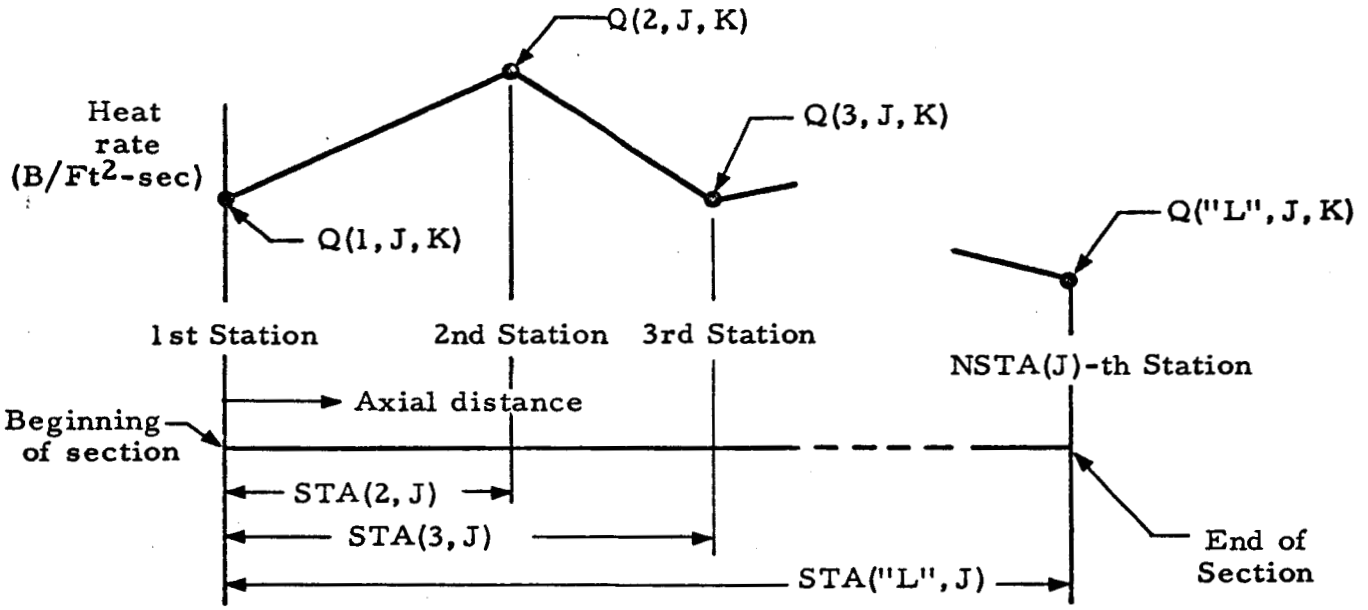
The range of I for the three preceding variables is 0, I or NS, depending on the value of INBC.

EXBC: External boundary condition control variable. In the present version of the SHORE program, this variable must always be non-zero (any arbitrary integer will suffice). In a later version of the program, it will control the exterior convection data input mode.

ENDBC(I): End boundary condition control variables for open ended shells (such as cylinders). If ENDBC (I) is set equal to zero, the program assumes that the I-th end surface is insulated (I = 1 → front end, and I = 2 → rear end of last section).

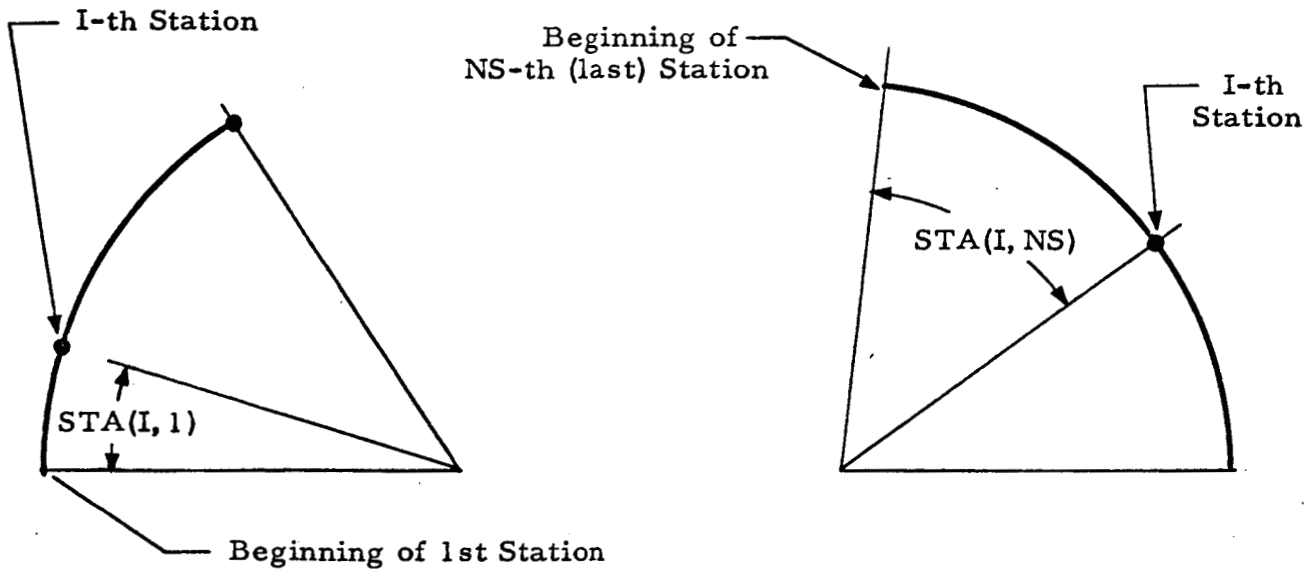
The present version of the input subroutine PREA allows only axisymmetric problems; although the rest of the program is fully configured for non-axisymmetric cases. "Convective" heat input rates are specified as piecewise-linear functions of time and surface locations by the variables NERA, ERA, NSTA, STA, and Q. NERA is the number of points in time at which convective rate distributions over the body are specified. The time at which the K-th distribution is given is ERA (K). The locations at which the convective rates are to be given are specified by the variables NSTA and STA, as shown on Figures 14 and 15 for non-spherical and spherical sections, respectively.

On Figure 14, the K-th heat rate distribution over the J-th section is indicated. In this case the number of "stations" in the section, NSTA (J), is indicated by "L".



Cylinders and Cones

Figure 14



(a) - Frontal Sphere (b) - Terminal Sphere

Spherical Sections

Figure 15

Heat input rates over the exterior surfaces of spheres are specified in the same way as the cylinders and cones; except that the "station locations", STA (_, _), refer to angular "distance" (radians) from the front of the section, rather than axial distance (in feet). In all cases, the first and last stations should coincide with the end points of the section. (Q and QEND - B/FT^2 - sec).

Heat input rates (if any) over the ends of "open" shells are also specified as piecewise-linear functions of time. QEND (I, J) is the heat rate at the I-th time, ERA (I). J = 1 → front end, and J = 2 → rear end of last section. The variables ENDEM(J) are end-surface emissivity values.

The program interpolates linearly in time and location for intermediate heat rates.

2.3.2 Internal Variables: A complete list of the internal variables (all variables other than those previously defined as "input variables") will not be undertaken since the significance of most of them is evident from the nature of their occurrence in the program. Certain variables which occur infrequently are defined directly in the flow charts. Other variables, grouped together according to function, are defined below.

2.3.2.1 General:

NW: is the number of "wedges" in the half-shell. In axisymmetric problems, NW is set equal to 1 by the program (in subroutine PREC) before the calculation of solutions is begun. (See sections 2.2.1 and 2.2.4).

NN: is the total number of bands; i. e. $\sum_{I=1}^{NL} MB(I)$.

NPD: is the number of nodes per disk (NN times NW).

NTØT: is the total number of nodes in the entire shell; including nodes of the "end caps", if present.

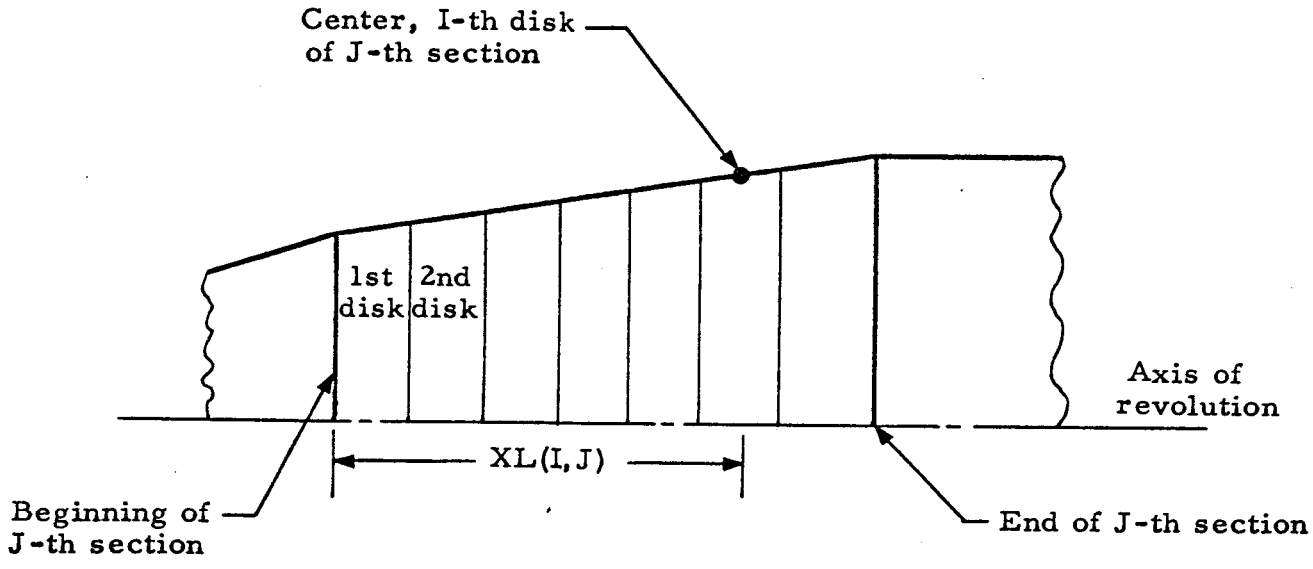
MD(I): is the number of disks in the I-th section. (See Section 2.2.1).

ØP(I): (logical)is true if and only if the I-th end of the shell is open (I = 1 → "nose", I = 2 → "tail").

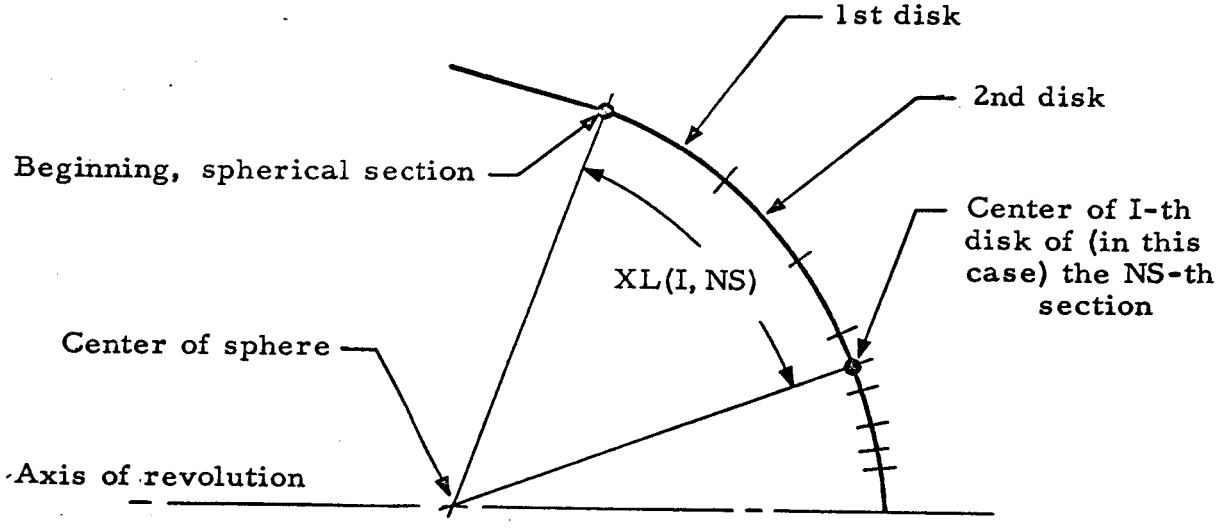
SPH(I): (logical) is true if and only if the I-th end section of the shell is spherical (I = 1 → "nose", I = 2 → "tail").

RADIUS (I) and SPHANG (I) are, respectively, the radius and the "angular length" of the I-th spherical end section (I = 1 → "nose", I = 2 → "tail"). For example, if the tail section is a complete hemisphere, SPHANG (2) = $\pi/2$.

XL (I, J): indicates the distance from the front of the J-th section to the center of its I-th disk, as shown on Figure 16.



Conical or cylindrical sections



Spherical sections

Figure 16

As depicted on Figure 16, XL (_, _) is either an axial distance or an angle measured positively from the front of the section toward the rear.

IE (_): is an "error trace matrix" employed in subroutines PREA and PREC to block execution of problems in which the limitations of dimension statements would be exceeded. If an error is detected, the program prints IE(_) to facilitate isolation of the error; then execution is terminated.

2.3.2.2 Layer and Band Radial Thicknesses:

S(I): is the radial thickness of the I-th surface layer element (see Section 2.2.3). Note that the S(_) are layer thickness terms-not band thicknesses.

Except for the outer layer, the layer thicknesses - hence the band thicknesses - within each layer of a particular disk do not vary with time and are constant around the shell circumference. The arrays ZS(_, _, _), DEND (_, _) and D(_) defined below represent band thicknesses within layers. The outer layer is accorded special treatment, as indicated in the flow charts for subroutine MARK.

ZS (I, J, K): is the radial thickness of all bands in the I-th layer of the J-th disk of the K-th spherical end section (K = 1 → "nose", K = 2 → "tail").

DEND(I, J): is the radial thickness of bands in the I-th layer of the J-th "end cap" (J = 1 → "nose", J = 2 → "tail").

D(I): is the radial thickness of all bands in the I-th layer of an arbitrary disk. The use of this variable is indicated in the flow chart for subroutine MARK. D(1) varies with "wedge" location, but D(2) through D(NL) are constant within each disk. D(1) is evaluated from the array S(_), and D(2) through D(NL) are obtained from either Z(_, _) or ZS(_, _, _); according to the type of section (spherical or non-spherical). Before subroutine MARK is entered, the array Z(_, _) is altered by subroutine PREC to indicate band, rather than layer, thicknesses. All Z(I, J) are divided by MB(I) except for I = 1.

2.3.2.3 Nodal Geometry: The array G(_, _) applies only to conical and cylindrical sections.

G(1, I): is the nodal squareness factor of the I-th section. It is constant for all nodes in the section. In the program output, this variable is labeled the "axial a/d" ratio (see Section 2.1.2).

G(2, I): is the "tangential a/d" ratio of all nodes in the I-th section. It is the ratio of the meridional length of a node to the circumferential distance between its center and the center of a tangentially-adjacent node.

The radial surface areas of all nodes in a given disk are the same.

G(3, I): is the radial surface area of nodes in the first disk of the I-th section.

G(4, I): is the "nodal area multiplier". The radial surface area of nodes in the J-th disk of the I-th section may be evaluated by multiplying the surface area of nodes in the (J-1)-th disk by G(4, I). This relation is a consequence of the nodal-subdivision method used by the program. In the special case of cylinders, G(4, _) is, of course, 1.0.

In the case of spherical end-sections, the two "a/d" ratios vary slightly from disk to disk; accordingly, the array GS(_, _, _) as defined below is used to represent nodal geometry. Where K = 1 → the "nose" section, and K = 2 → "tail" section, nodal properties in the J-th disk are:

GS(1, J, K): is the "squareness factor" or "axial a/d" ratio,

GS(2, J, K): is the "tangential a/d" ratio, as defined for G(2, _), and

GS(3, J, K): is radial surface area.

AE(K) and FAE(K): are the radial surface area and the effective "squareness factor", respectively, of nodes in the K-th "end cap" (K = 1 → "nose", K = 2 → "tail").

2.3.2.4 Exterior Boundary Conditions: The exterior boundary condition input data read by subroutine PREA is operated on by subroutine PREC to generate a file of standard "heat input records" on a magnetic tape. Each such record represents the heat input rates over the exterior surface of the shell at a specific time. The intervening time period between an adjacent pair of records is called an "era". During execution of a solution, subroutine MARK retrieves heat input records (one at a time) such that it always has two complete records in core storage; one for the beginning of the current era, and one for the end. When the end of a particular era is reached, it is necessary to read just one additional record, since the end of one era is the beginning of the next. Components of standard heat input records are defined below, using the FORTRAN variable names appearing in subroutine MARK. The index JQ appearing in these definitions, which may be either 1 or 2, indicates that the associated variables apply to the end of an era. Another variable, IQ, is always

associated with data applying to the beginning of an era. At the end of each era, the values of IQ and JQ are alternated before the next heat record is read.

LAST is a logical variable indicating (if .TRUE.) that the associated record is the last one in the file (announcing the end of the problem).

HOTWAL: is a logical variable indicating (if .TRUE.) that "hot wall coefficients", as defined below, are included in the associated record.

R(I, JQ): for I = 1 through NWEDGE, is the "cold wall" heat input rate at the I-th "surface node" location. "Surface node" indexing order is essentially the same as for "surface layer elements"; i.e. the first NW surface nodes are in the first disk of the shell, the (NW + 1)-th through the (2*NW)-th are in the second disk etc., in serial order corresponding to wedge location.

REND(K, JQ) (K = 1 → "nose", K = 2 → "tail") are the cold wall rates at the open edges or end caps, as the case may be.

C(I, JQ) and CEND(K, JQ): are "hot wall" coefficients at locations corresponding to R(I, JQ) and REND(K, JQ), respectively.

ERA(JQ): is the time at which the foregoing variables apply.

DERA = ERA(JQ) - ERA(IQ): is computed by subroutine MARK at the beginning of each era. It is the time duration of the era.

TERA: which is incremented by DT at each step in the solution, is the time elapsed since the beginning of the current era.

"Cold wall" heat rates (and "hot wall" coefficients, if applicable) for all surface nodes are evaluated at each step in the solution process (i.e. for each TERA) by linear-interpolation (in time).

Q(I): the heat input rate at the I-th surface node is evaluated by MARK as an interpolated "cold wall" rate plus, if applicable, the temperature of the node times the interpolated "hot wall" coefficient. The heat rates at the end caps (or edges) are computed similarly and stored in the array QEND(_).

2.3.2.5 Material Properties at Individual Nodes: When, during execution of the solution process, material properties at a particular set of nodes are required, subroutine MARK calls subroutine PROP to evaluate the properties and return the results via the COMMON variables PR(_, _, _) and EMIS(_, _). The implicit arguments of PROP are LW, IZP, ISP, and JB. During a single "call", PROP evaluates properties at all nodes within a particular disk

or "end cap", depending upon the arguments. MARK sometimes requires two such arrays of property data at the same time (e.g. when computing axial conduction terms across a section interface); accordingly, the variable JB, which may be either 1 or 2, is used to control storage assignment.

PR(1, J, JB): is thermal conductivity at the J-th node in the disk (or end cap).

PR(2, J, JB): is specific heat at the J-th node in the disk (or end cap).

EMIS(K, JB): is emissivity at the K-th surface node within the disk. The range of the dummy variables J and K above is determined by the argument LW. Subroutine MARK sets LW equal to NW or 1 according to whether properties are required for a disk or an end cap, respectively. The range of J is 1 through LW*NN and the range of K is 1 through LW.

ISP: is the section in which the disk, or end cap, is located.

IZP: indicates the applicable temperature; i.e. PR(I, J, JB) is evaluated using the temperature T(IZP + J).

2.3.2.6 Temperature and Heat Accumulation Arrays:

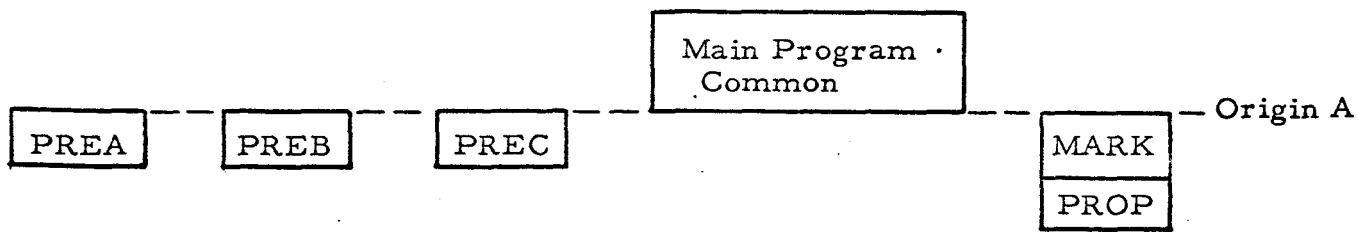
T(I): is the temperature of the I-th node in the shell (see Section 2.2).

U(J): is the "heat accumulation" for the J-th node within a particular disk (see Section 2.2).

QA(J): is the rearwards axial heat conduction term at the J-th node of a particular disk (see Section 2.2).

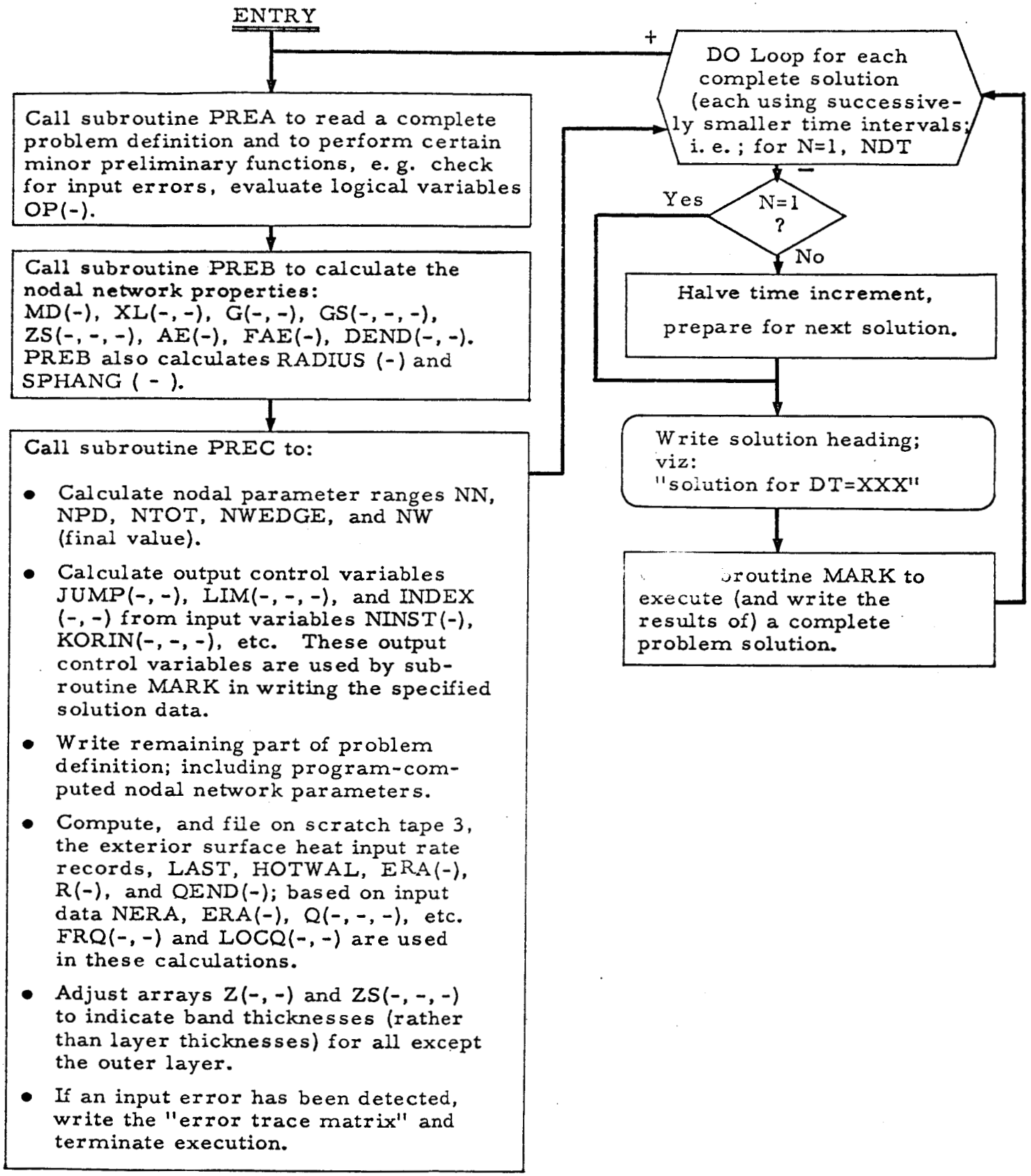
TSEC(I) and QSEC(I): are the temperature and "heat accumulation", respectively, for the I-th heat sink; i.e. the heat sink on the interior of the I-th section.

2.4 Flow Charts: The SHORE program consists of a main program and five subroutines. In order to provide sufficient core storage for the analysis of very "large" problems, the OVERLAY structure shown on Figure 17 is used.



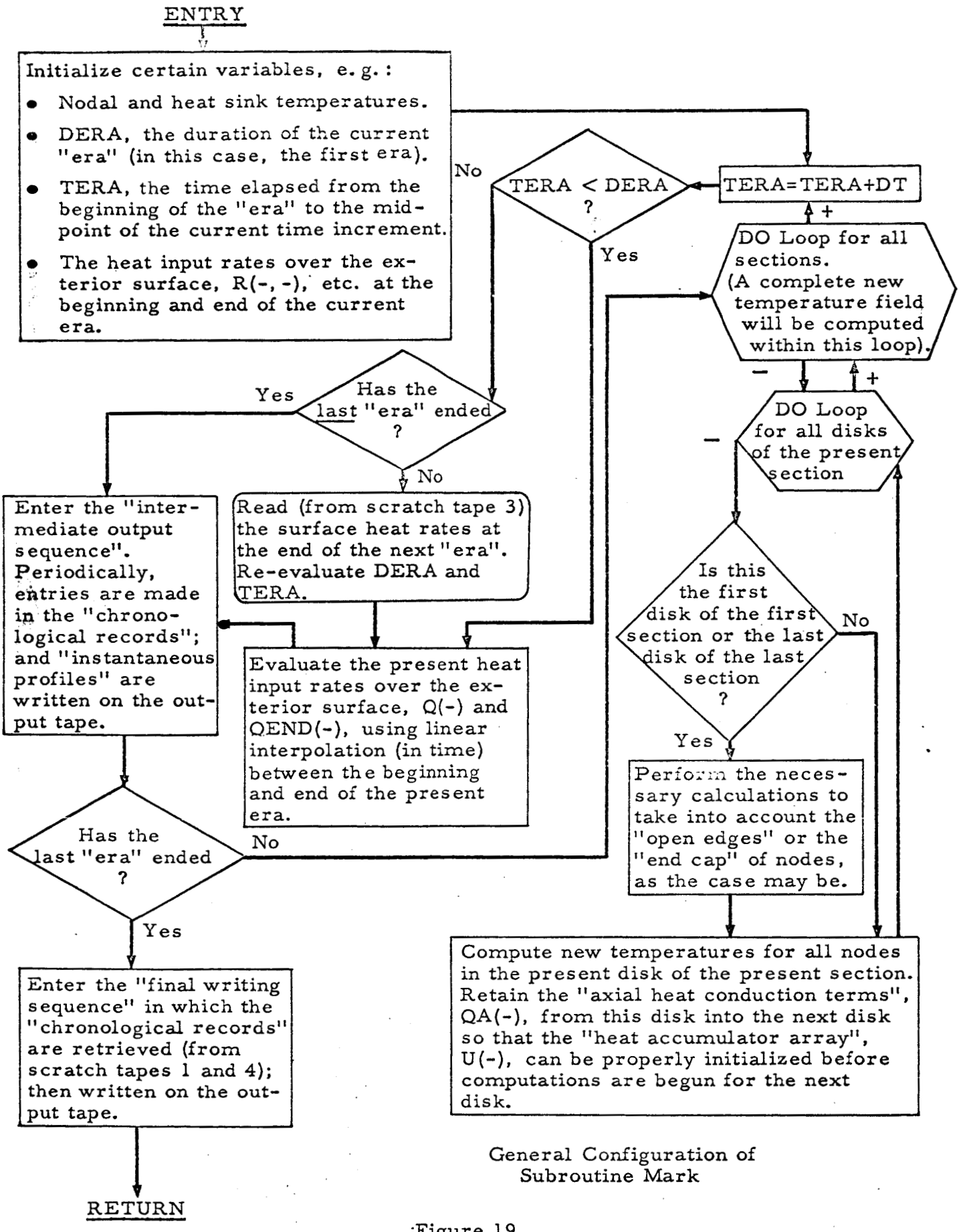
Overlay Structure

Figure 17



Main Program Flow Chart

Figure 18



General Configuration of Subroutine Mark

Figure 19

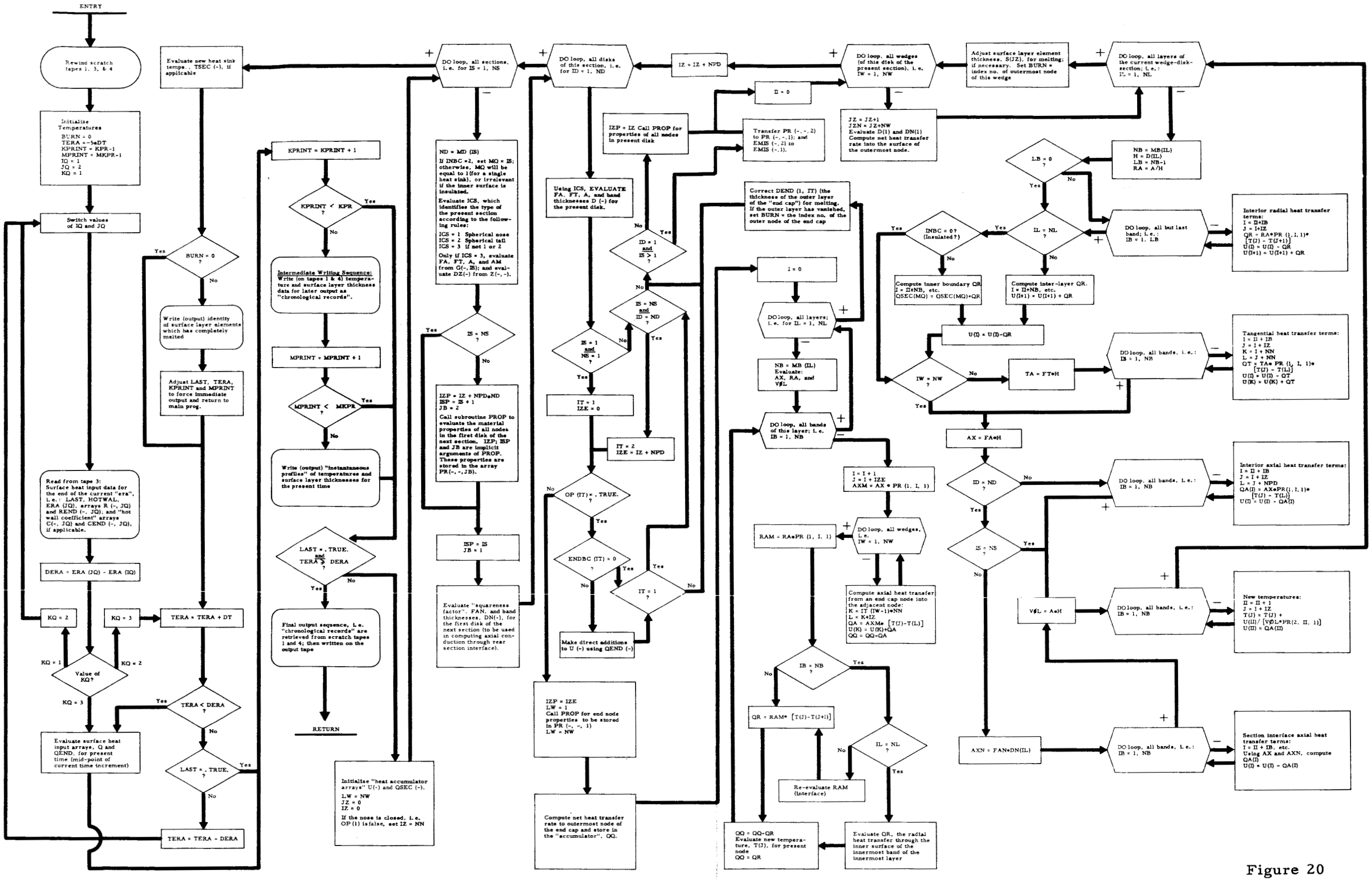

ENTRY
Rewind scratch tapes 1, 3, & 4
Initialize Temperatures
BURN = 0
TERA = -5*DT
KPRINT = KPR-1
MPRINT = MKPR-1
JQ = 1
JQ = 2
KQ = 1
Switch values of IQ and JQ
Read from tape 3: Surface heat input data for the end of the current "era", i.e.: LAST, HOTWAL, ERA (JQ), arrays R (-, JQ) and REND (-, JQ), and "hot wall coefficient" arrays C(-, JQ) and CEND (-, JQ), if applicable.
DERA = ERA (JQ) - ERA (IQ)
KQ = 2
KQ = 3
TERA = TERA + DT
Value of KQ?
KQ = 1
KQ = 2
KQ = 3
TERA < DERA ?
Yes
No
Evaluate surface heat input arrays, Q and QEND, for present time (mid-point of current time increment)
LAST = . TRUE. ?
Yes
No
TERA = TERA - DERA
Evaluate new heat sink temps., TSEC (-), if applicable
BURN = 0 ?
Yes
No
Write (output) identity of surface layer elements which has completely melted
Adjust LAST, TERA, KPRINT and MPRINT to force immediate output and return to main prog.
KPRINT = KPRINT + 1
KPRINT < KPR ?
Yes
No
Intermediate Writing Sequence: Write (on tapes 1 & 4) temperature and surface layer thickness data for later output as "chronological records".
MPRINT = MPRINT + 1
MPRINT < MKPR ?
Yes
No
Write (output) "instantaneous profiles" of temperatures and surface layer thicknesses for the present time
LAST = . TRUE. and TERA ≥ DERA ?
Yes
No
Final output sequence, i.e. "chronological records" are retrieved from scratch tapes 1 and 4; then written on the output tape
RETURN
DO loop, all sections, i.e. for IS = 1, NS
ND = MD (IS)
If INBC ≠ 2, set MQ = IS; otherwise, MQ will be equal to 1 (for a single heat sink), or irrelevant if the inner surface is insulated.
Evaluate ICS, which identifies the type of the present section according to the following rules:
ICS = 1 Spherical nose
ICS = 2 Spherical tail
ICS = 3 If not 1 or 2
Only if ICS = 3, evaluate FA, FT, A, and AM from G(-, IS); and evaluate DZ(-) from Z(-, -).
IS = NS ?
Yes
No
IZP = IZ + NPD*ND
ISP = IS + 1
JB = 2
Call subroutine PROP to evaluate the material properties of all nodes in the first disk of the next section. IZP, ISP and JB are implicit arguments of PROP. These properties are stored in the array PR(-, -, JB).
ISP = IS
JB = 1
Evaluate "squareness factor", FAN, and band thicknesses, DN(-), for the first disk of the next section (to be used in computing axial conduction through rear section interface).
Initialize "heat accumulator arrays" U(-) and QSEC (-).
LW = NW
JZ = 0
IZ = 0
If the nose is closed, i.e. OP (1) is false, set IZ = NN
DO loop, all disks of this section, i.e. for ID = 1, ND
IZ = IZ + NPD
IZP = IZ. Call PROP for properties of all nodes in present disk
Using ICS, EVALUATE FA, FT, A, and band thicknesses D (-) for the present disk.
ID = 1 and IS > 1 ?
No
Yes
IS = 1 and NS = 1 ?
No
Yes
IS = NS and ID = ND ?
No
Yes
IT = 1
IZE = 0
IT = 2
IZE = IZ + NPD
OP (IT) = . TRUE. ?
No
Yes
ENDBC (IT) = 0 ?
No
Yes
IT = 1 ?
No
Yes
Make direct additions to U (-) using QEND (-)
IZP = IZE
LW = 1
Call PROP for end node properties to be stored in PR (-, -, 1)
LW = NW
Compute net heat transfer rate to outermost node of the end cap and store in the "accumulator", QQ.
II = 0
Transfer PR (-, -, 2) to PR (-, -, 1); and EMIS (-, 2) to EMIS (-, 1).
DO loop, all wedges (of this disk of the present section), i.e. IW = 1, NW
JZ = JZ+1
JZN = JZ+NW
Evaluate D(I) and DN(I)
Compute net heat transfer rate into the surface of the outermost node.
Correct DEND (1, IT) (the thickness of the outer layer of the "end cap") for melting. If the outer layer has vanished, set BURN = the index no. of the outer node of the end cap
I = 0
DO loop, all layers; i.e. for IL = 1, NL
NB = MB (IL)
Evaluate:
AX, RA, and V∯L
DO loop, all bands of this layer; i.e. IB = 1, NB
I = I + 1
J = I + IZE
AXM = AX * PR (I, I, 1)
RAM = RA*PR (I, I, 1)
DO loop, all wedges, i.e. IW = 1, NW
Compute axial heat transfer from an end cap node into the adjacent node:
K = IT (IW-1)*NN
L = K+IZ
QA = AXM* [T(J)-T(L)]
U(K) = U(K)+QA
QQ = QQ-QA
IB = NB ?
No
Yes
QR = RAM* [T(J)-T(J+1)]
IL = NL ?
No
Yes
Re-evaluate RAM (interface)
QQ = QQ-QR
Evaluate new temperature, T(J), for present node
QQ = QR
Evaluate QR, the radial heat transfer through the inner surface of the innermost band of the innermost layer
Adjust surface layer element thickness, S(IZ), for melting; if necessary. Set BURN = index no. of outermost node of this wedge
DO loop, all layers of the current wedge-disk-section; i.e.: IL = 1, NL
NB = MB(IL)
M = D(IL)
LB = NB-1
RA = A/H
LB = 0 ?
Yes
No
IL = NL ?
Yes
No
INBC = 0? (Insulated?)
Yes
No
Compute inner boundary QR
I = II+NB, etc.
QSEC(MQ) = QSEC(MQ)+QR
Compute inter-layer QR.
I = II+NB, etc.
U(I+1) = U(I+1) + QR
DO loop, all but last band; i.e.: IB = 1, LB
Interior radial heat transfer terms:
I = II+IB
J = I+IZ
QR = RA*PR (I, I, 1)* [T(J) - T(J+1)]
U(I) = U(I) - QR
U(I+1) = U(I+1) + QR
U(I) = U(I)-QR
IW = NW ?
No
Yes
TA = FT*H
DO loop, all bands, i.e.: IB = 1, NB
Tangential heat transfer terms:
I = II + IB
J = I + IZ
K = I + NN
L = J + NN
QT = TA* PR (I, I, 1)* [T(J) - T(L)]
U(I) = U(I) - QT
U(K) = U(K) + QT
AX = FA*H
ID = ND ?
No
Yes
DO loop, all bands, i.e.: IB = 1, NB
Interior axial heat transfer terms:
I = II + IB
J = I + IZ
L = J + NPD
QA(I) = AX*PR(I, I, 1)* [T(J) - T(L)]
U(I) = U(I) - QA(I)
IS = NS ?
Yes
No
V∯L = A*H
DO loop, all bands, i.e.: IB = 1, NB
New temperatures:
II = II + 1
J = I + IZ
T(J) = T(J) + U(II)/ [V∯L*PR(2, II, 1)]
U(II) = QA(II)
AXN = FAN*DN(IL)
DO loop, all bands, i.e.: IB = 1, NB
Section interface axial heat transfer terms:
I = II + IB, etc.
Using AX and AXN, compute QA(I)
U(I) = U(I) - QA(I)

Figure 20

A flow chart for the main program is presented on Figure 18. The "preliminary" subroutines, PREA, PREB and PREC (the functions of which are described on Figure 18) are comparatively straightforward; accordingly, flow charts are not presented for them.

Two charts are presented for the primary computational subroutine, MARK. The first chart (Figure 19) briefly presents the general organization of subroutine MARK (it is not actually a flow chart of the coded instructions). A detailed flow chart of this routine is given on Figure 20.

Subroutine MARK frequently calls PROP to evaluate the material properties of a collection of nodes (either all nodes in a particular disk or all nodes in an "end cap"). Since subroutine PROP is very short, no flow chart is given for it.

2.5 Sample Problem: A list of the SHORE program is appended to this report. An execution of the sample problem described below follows the list.

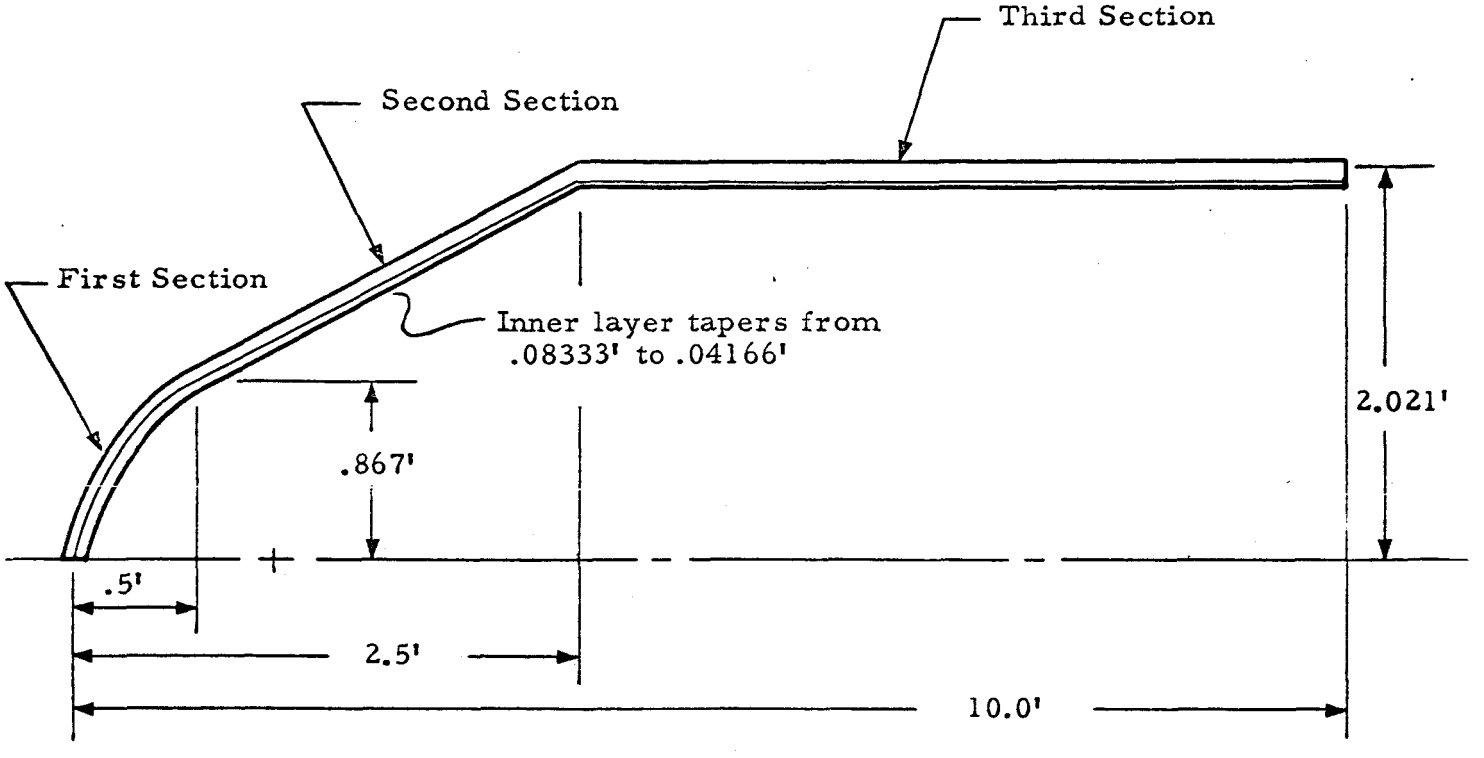


Figure 21

The overall geometry of the sample problem is as shown on Figure 21. The outer layer is of constant thickness (.08333 ft.)

in all sections. The inner layer is .08333 ft. thick throughout the first (spherical nose) section; then it tapers linearly to .04166' at the end of the second section and remains constant (.04166 ft.) through the last section. The outer layer is composed of a (ficticious) material we will call FIBERGLAS-A; the inner layer for all sections is of FIBERGLAS-B.

The properties of these two materials are as indicated in the following table:

Material number (arbitrarily assigned)	1	2
Name	FIBERGLAS-A	FIBERGLAS-B
Density	115.4	107.4
Melting temperature	1200.	—
Latent heat	30.	—
Thermal conductivity	1.28×10^{-4}	$.944 \times 10^{-4}$
Specific heat	.283	.362
Emissivity	.3	—

The above properties are constant (i. e. not temperature-dependent). The units are Btu-^{o}R-lb. -ft-sec.

The problem is axisymmetric, with the interior and rear-edge surfaces insulated.

The heat-input rates over the exterior surface are as indicated on Figure 22. Rates at all locations are linear functions of time.

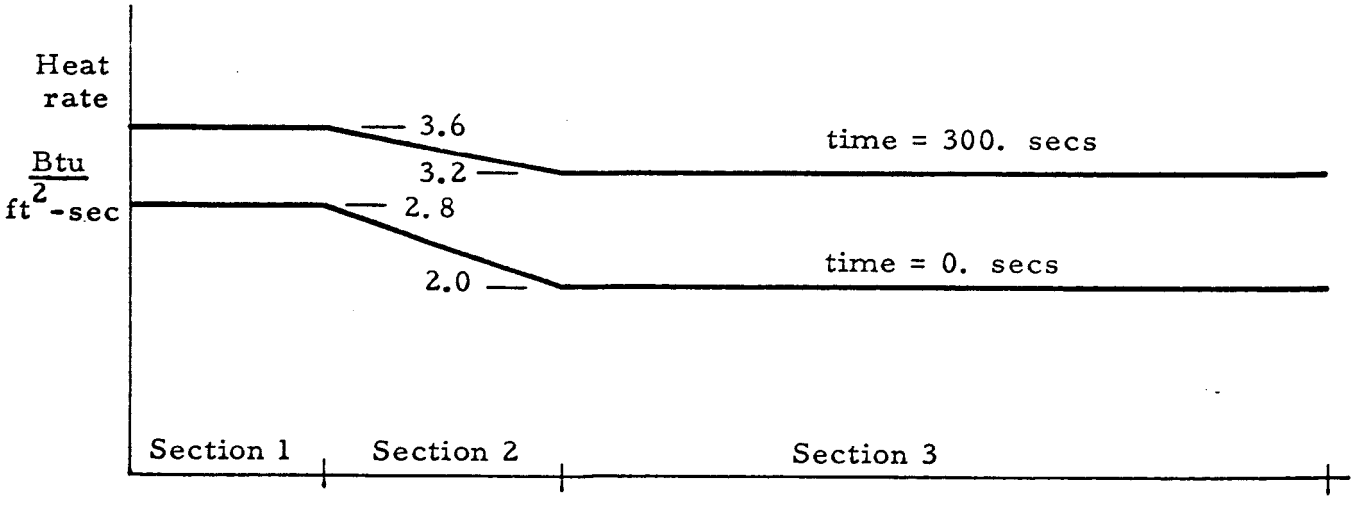


Figure 22

The initial temperature of the body is 600°R and the ambient temperature (for purposes of computing radiant heat losses) is 0. °R.

The foregoing essentially completes the problem definition, except for the numerical solution parameters which are discussed below.

Each of the two layers is divided into five bands. Six tangential nodal subdivisions (wedges) will be used. Squareness factors of 1.0 will apply in each of the three sections. Two complete solutions of this problem will be effected; one using a time increment of 5.0 secs, the other (half the original) 2.5 secs.

The data cards corresponding to this execution are listed on pages 38 and 39.

The nodal geometry computed by the program appears as shown on Figure 23.

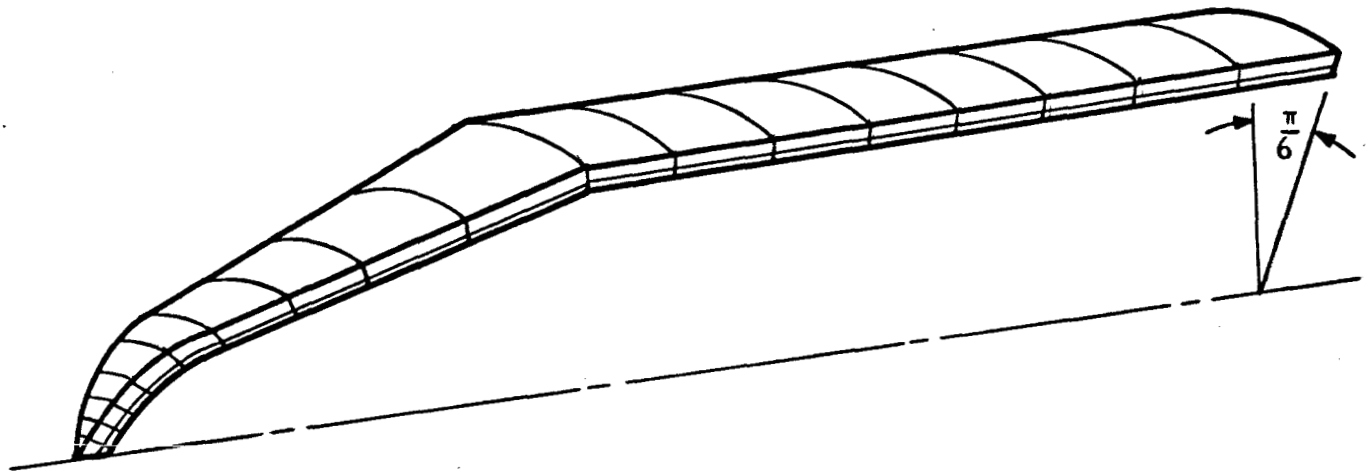


Figure 23

The radial nodal subdivisions (five in each of the two layers) are not shown on Figure 23.

The "local axial coordinates" of disks appearing in the output are defined in Section 2.3.2. These coordinates (either radians in the case of spherical sections, or distances for cones and spheres) specify the locations of the disk centers.

The output is mostly self-explanatory, except for certain items (e.g. FRQ, LOCQ, etc.) which are normally not important to

users of the program. The E - conversion solution profiles are surface layer thicknesses; the F - conversions are temperatures. Elements of the profile vectors are ordered positively from the outer band toward the inner bands, the first disk toward the last disk, etc.

COMPOSITE BODY(SPHERE,CONE,CYLINDER) MAY 20,1964

	5.+0		6.+2		0.+0				
2	1	20	1	13	3	10	10		
1	1	1		3					
1	3	1		3					
1	6	1	50	1					
1	9	1		3					
2	1	1		3					
2	2	1		3					
2	6	1	50	1					
2	10	1		3					
3	1	1		3					
3	2	1		3					
3	5	1		3					
3	6	1	50	1					
3	10	1		3					
1		1		3					
2		1		3					
3		1		3					
1	1	1	10						
1	2	1	29						
1	1	1	81						
2	1	1	29						
2	2	1	29						
2	1	1	53						
2	2	1	53						
3	1	1	50						
3	5	1	50						
3	10	1	50						
1	1	1	10						
1	1	1	17						
1	1	1	29						
1	1	1	49						
1	1	1	81						
2	1	1	9						
2	1	1	29						
2	1	1	53						
2	1	1	84						
3	1	1	6						
3	2	6	1	0	5	5			
1	2								
1	2								
1	2								
2	0								
1	FIBERGLAS-A			1.154+2		1.2+3		3.+1	2
	7.+1		1.28-4		2.83-1		3.-1		
	5.+4		1.28-4		2.83-1		3.-1		
2	FIBERGLAS-B			1.074+2		1.2+3		1.+3	2
	7.+1		9.44-5		3.62-1				
	5.+4		9.44-5		3.62-1				
	1.+0		1.+0		1.+0				
	0.+0		0.+0		8.333-2		8.333-2		
	5.-1		8.67-1		8.333-2		8.333-2		

(Continued)

```
 2.5+0    2.021+0    8.333-2    4.166-2
 10.+0    2.021+0    8.333-2    4.166-2

0    1
2
  0.+0       3.+2
2
  0.+0       5.-1
 2.8+0      2.8+0
 3.6+0      3.6+0
2
  0.+0       2.+0
 2.8+0       2.+0
 3.6+0      3.2+0
2
  0.+0       7.5+0
  2.+0       2.+0
 3.2+0      3.2+0
```

3. COMMENTS

The SHORE program was designed to permit economical analysis of problems involving a large number of nodes. Most transient heat conduction programs of which the writer is aware are limited to a few hundred nodes. The present version of the SHORE program admits up to 5,000 nodes (this limit can be expanded considerably, if required).

In analyzing particular problems, it is clearly desirable in many cases to effect several different numerical solutions employing successively "finer" nodal networks and, for each network, successively smaller solution time increments. In this way, one can observe the effect of the numerical solution parameters on the predicted temperature fields (i.e., convergence of the sequence of numerical processes).

Understandably, one might ask why a constant solution time increment is used, rather than having the program automatically adjust the time increment during evaluation of the solution (based on "stability criteria", e.g. $\frac{1}{2}\frac{\rho c}{k}(\Delta x)^2$, etc). The reason is that such an automatic process would require a significant amount of additional computation (hence, increase execution time), particularly if carried out at each time increment; i.e., to compute "new time" increments almost as much work would be required as to compute "new" temperatures at each step in the solution process. Satisfying the "stability criteria" assures that the numerical process will not produce obviously erroneous results (e.g., "wild" temperature fluctuations in violation of the 2nd law of Thermodynamics). However, even if the stability criteria are satisfied, there is no guarantee that a suitably accurate approximation to the actual transient solution has been obtained. The writer knows of no general criteria for determining how small the time increments and nodal dimensions must be to assure that a stated degree of accuracy (however that might be defined) is achieved. Accordingly, successive solutions, as described in the preceding paragraph, appear to be an appropriate approach to the problem.

Throughout the program, a number of operations are based upon assumptions of small nodal dimensions and short time increments consistent with the linearized assumptions (e.g., linear approximations of temperature space derivatives) implicit in the general numerical solution method. For example, material properties are evaluated on the basis of node-center temperatures rather than on space-average temperatures or some other complex approximation, because such a "refinement" would incur a substantial penalty in execution time while contributing negligibly to solution accuracy. By widely employing assumptions of this type, a relatively high degree of computational efficiency is achieved by the program, so that analysis of comparatively "fine" nodal networks is economically feasible.

While the SHORE program was being checked-out, certain problems for which "exact" solutions were known were analyzed. In these cases, it was observed that although refinement of the nodal network improved the results (mainly in that it gave a more nearly complete "picture" of the temperature distribution), reduction of time increment length below about half of the "stability criteria" - limited upper bound did not significantly improve the results, which were already quite satisfactory.

It is the writer's opinion that general-purpose programs such as the one presently described should, in general, be based upon the simplest approximations (i. e., those requiring the least arithmetic computation), which would necessarily approach the exact solution as "increment" size (e. g. nodal dimensions and time increment length) approaches zero. In this way, it becomes practical in terms of computer execution cost to employ relatively small "increments", affording closer approximations to certain critical quantities (e. g. temperature derivatives) than could be achieved with larger "increments", regardless of the arithmetic process details.

It is anticipated that the SHORE program will occasionally be modified. Comparatively minor modifications would be required to provide for the following:

- Orthotropic materials (e. g. pyrolitic graphite).
- More sophisticated treatment of surface layer erosion (e. g. "charring").
- More elaborate output (e. g. plots).
- Automatic determination of time increment length; i. e., re-evaluation of the time increment after (say) an arbitrary number of solution steps have taken place, but <u>not</u> at every step for reasons previously cited.
- More general geometry and nodal network topology; e. g. different numbers of layers in different sections, "thick" shells.

APPENDIX

List of Computer Program

and

Sample Problem Execution

(Bound Separately)